



**A University of Sussex PhD thesis**

Available online via Sussex Research Online:

<http://sro.sussex.ac.uk/>

This thesis is protected by copyright which belongs to the author.

This thesis cannot be reproduced or quoted extensively from without first obtaining permission in writing from the Author

The content must not be changed in any way or sold commercially in any format or medium without the formal permission of the Author

When referring to this work, full bibliographic details including the author, title, awarding institution and date of the thesis must be given

Please visit Sussex Research Online for more information and further details

# Modelling and simulations of a viscous model for cell migration

**BENARD KIPCHUMBA KIPLANGAT**

Submitted for the degree of Doctor of Philosophy

University of Sussex

December 2018

# Declaration

I hereby declare that this thesis has not been and will not be submitted in whole or in part to another University for the award of any other degree.

Signature:

BENARD KIPCHUMBA KIPLANGAT

UNIVERSITY OF SUSSEX

BENARD KIPCHUMBA KIPLANGAT, DOCTOR OF PHILOSOPHY

MODELLING AND SIMULATIONS OF A VISCOUS MODEL FOR CELL MIGRATION

SUMMARY

This thesis presents a mathematical and computational model for cell migration that couples a system of reaction-advection-diffusion equations describing the interactions between F-actin and myosin II to a force balance equation describing the velocity vector of the actin-myosin network.

Cell migration plays a crucial role in many biological processes. In eukaryotic cells, this migration is largely powered by a system of actin and myosin II. At the leading edge of the cell, cross-linked actin filaments polymerise by adding actin monomers to their ends while at the back of the cell, myosin II binds to a bundle of actin filaments. These processes create protrusive and contractile forces generated by the action of actin polymerisation and myosin II contraction.

Based on the idea that cell migration is powered by the actin-myosin network of the cell, we formulate model equations for migrating cell which comprises reaction-advection-diffusion equations that are coupled to a force balance equation describing the velocity vector of the network. This is a viscous model with active stresses coming from the actin-myosin system. These equations describing the migrating cells are highly nonlinear partial differential equations with no closed form solutions and we therefore resort to numerical methods in order to compute the approximate solution.

F-actin and myosin II solution are the solution for the reaction-advection-diffusion equations while the speeds of the cell come from the solution of the force balance equation. We begin simulations on a unit disk at zero initial velocity with different data for the initial conditions of F-actin and myosin II concentrations. We also vary some parameters at a time while keeping all the other parameters constant: for example (i) total amount of actin  $\rho_a^{tot}$  and (ii) contraction coefficient for myosin II  $\eta_m^0$ .

Actin polymerisation causes protrusive stress at the cell periphery which results in expansion of the cell. We observe that the initial conditions play an important role in the spatiotemporal dynamics of F-actin as well as the evolution of the cell shape. Actin changes from the active state (F-actin) to inactive state (G-actin) and vice-versa through polymerisation and depolymerisation processes and hence the total amount of actin is conserved at any time. We note that in our model, myosin II only diffuses inside the cell and exerts contractile stress in the cell. Its total concentration in the entire cell is conserved.

# Acknowledgements

I would like to acknowledge and thank my supervisor Professor Anotida Madzvamuse for his inspiration, guidance and invaluable advice throughout my research. His encouragement, support and patience led to the successful completion of this thesis. I am forever grateful to the University of Sussex for fully funding my research and stay at the university through the Chancellor's International Research Scholarship. This provided me a conducive environment to carry out research. Thanks also to Dr. Chandrasekhar Venkataraman for his co-supervision and advice. I also thank all my colleagues in the Mathematics department particularly Wakil Sarfaraz, Muflih Alhazmi, Eduard Campillo-Funollet, Laura Murphy, Victor Juma and Davide Cusseddu for walking through this very wonderful research journey together and for all the insightful discussions on our research. Thanks to all the support staff of the School of Mathematics and Physical Sciences (MPS) especially Richard Chambers and Rebecca Foster for being so helpful and providing the assistance I needed throughout my studies. Special thanks to my wife Sharon Chepkemai, parents, brothers and sisters for their support and encouragement.

# Contents

<b>List of Tables</b>	<b>vii</b>
<b>List of Figures</b>	<b>x</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Cell structure and function . . . . .	1
1.1.1 Structure of eukaryotic cell . . . . .	1
1.1.2 Actin cytoskeleton and force generation . . . . .	4
1.1.3 Cell adhesion . . . . .	6
1.1.4 Signaling pathways . . . . .	7
1.2 Models for cell migration . . . . .	8
1.3 Numerical methods for models of cell migration . . . . .	10
1.4 Outline of the thesis . . . . .	11
<b>2 Model formulation for cell migration</b>	<b>13</b>
2.1 Introduction . . . . .	13
2.2 Mathematical preliminaries . . . . .	13
2.3 Derivation of viscous model for cell migration . . . . .	15
2.4 Phase-field model . . . . .	25
2.5 Summary . . . . .	30
<b>3 Numerical methods for the viscous model for cell migration</b>	<b>32</b>
3.1 Introduction . . . . .	32
3.2 The finite element method . . . . .	33
3.2.1 Theory of the finite element method . . . . .	33
3.2.2 The Schnakenberg model . . . . .	36
3.2.3 The viscous model for cell migration . . . . .	50
3.3 Finite differences for the phase-field model . . . . .	88
3.3.1 Theory of finite difference method . . . . .	88
3.3.2 Finite differences for the phase-field model for cell migration . . . . .	90
3.3.3 Numerical simulation . . . . .	99
3.3.4 Summary . . . . .	99

<b>4 Summary and future directions</b>	<b>101</b>
4.1 Summary . . . . .	101
4.2 Future directions . . . . .	103
<b>Bibliography</b>	<b>105</b>

# List of Tables

2.1	Dimensional parameters and their values. . . . .	22
2.2	Non-dimensional parameter values for the non-dimensionalised viscous model. . . .	24
2.3	Non-dimensionalised parameters for the phase-field model. . . . .	30
3.1	Convergence of the $u$ variable using the 2-SBDF scheme at different time steps $\tau$ showing the effects of time-step refinement on the magnitudes of errors. . . . .	47
3.2	The parameters for the Schnakenberg model. . . . .	49
3.3	Non-dimensional parameters for the non-dimensionalised viscous model . . . . .	61
3.4	The numerical normals to a circle converge to the exact normal to a circle as the mesh is refined. . . . .	76
3.5	The nondimensionalised values of the parameters used in the model. . . . .	90



# List of Figures

1.1	Schematic representation of the cell showing the components of the cell Pullarkat et al. (2007). . . . .	3
1.2	Schematic representation of actin polymerisation and depolymerisation Pollard and Borisy (2003) . . . . .	5
2.1	Double well potential for the phase-field. . . . .	27
3.1	The canonical quadrilateral element $\hat{K}$ . . . . .	42
3.2	A general quadrilateral element $K$ . . . . .	43
3.3	Transformation of a general quadrilateral into the canonical quadrilateral element. . . . .	43
3.4	Solutions for the $u$ variable of the Schnakenberg model using a 2-SBDF scheme with $\tau = 2 \times 10^{-3}$ . Blue signifies lowest values while red highest values. (a) Initial condition as random perturbations about steady states for the $u$ variable and (b) solution at a final time step $t = 10$ showing convergence to an in-homogeneous steady state. Parameters values used are $a = 0.1$ , $b = 0.9$ , $d = 10$ and $\gamma = 29$ . . . . .	46
3.5	Solutions for the $v$ variable of the Schnakenberg model using a 2-SBDF scheme with $\tau = 2 \times 10^{-3}$ . Blue signifies lowest values while red highest values. (a) Initial condition as random perturbations about steady states for the $v$ variable and (b) solution at a final time step $t = 10$ showing convergence to an in-homogeneous steady state. Parameters values used are $a = 0.1$ , $b = 0.9$ , $d = 10$ and $\gamma = 29$ . . . . .	47
3.6	Solutions for the $u$ variable of the Schnakenberg model with $\tau = 2 \times 10^{-3}$ , $a = 0.1$ , $b = 0.9$ , $d = 10$ and (a) $\gamma = 29$ and (b) $\gamma = 100$ showing convergence to in-homogeneous steady states. Blue signifies lowest values while red highest values. . . . .	48
3.7	Convergence history of the simulations of the Schnakenberg model using a 2-SBDF scheme for the $u$ variable with time refinements. . . . .	48
3.8	Convergence history of the simulations of the Schnakenberg model for the $v$ variable with mesh refinements. . . . .	49
3.9	Triangulation of the unit disk using quadrilateral elements after (a) two global mesh refinements, (b) three global mesh refinements, (c) four global mesh refinements and (d) five global mesh refinements showing convergence to the unit disk. . . . .	57

3.10	Solutions for the $\rho_m$ variable on a stationary unit disk using $\tau = 2 \times 10^{-3}$ . Blue signifies the lowest values while red the highest values. (a) Initial condition for myosin II (b) $\rho_m$ solution at time $t = 0.8$ and (c) $\rho_m$ solution at time $t = 3$ showing convergence to a homogeneous steady state. . . . .	58
3.11	Solutions for the $\rho_a$ variable on a stationary unit disk using $\tau = 2 \times 10^{-3}$ . Blue signifies the lowest values while red the highest values. (a) Initial condition for F-actin (b) $\rho_a$ solution at time $t = 0.8$ and (c) $\rho_a$ solution at time $t = 3$ showing convergence to a homogeneous steady state. . . . .	58
3.12	(a) Conservation of mass for actin and (b) diffusion coefficient for myosin II. We used time-step $\tau = 2 \times 10^{-3}$ and mesh size $h = 0.055126$ . . . . .	59
3.13	Convergence of the $\rho_a$ variable with different time steps $\tau$ . . . . .	59
3.14	Graphical displays of the numerical results of the myosin II concentration $\rho_m$ using a 2-SBDF scheme with $\tau = 0.001$ . Blue signifies lowest values while red highest values. (a) Initial condition as random perturbation about $\rho_m = 1.0$ (b) $\rho_m$ at time $t = 0.004$ , (c) $\rho_m$ at time $t = 2$ and (d) solution at final time $t = 4$ . . . . .	78
3.15	Graphical displays of the numerical results of the F-actin concentration $\rho_a$ using a 2-SBDF scheme with $\tau = 0.001$ . Blue signifies lowest values while red highest values. (a) Initial condition as non-zero only in one half of the cell (b) $\rho_a$ at time $t = 0.004$ , (c) $\rho_a$ at time $t = 2$ and (d) solution at final time $t = 4$ . . . . .	79
3.16	Graphical display of the speed of the cell as the solution of the force balance equation for the cell. Blue signifies lowest values while red highest values. (a) Cell at initial stationary state (b) cell speed at time $t = 0.004$ , (c) cell speed at time $t = 2$ , (d) cell speed at final time $t = 4$ and (e) area of the evolving cell as a function of time showing increase in the area. . . . .	80
3.17	Solutions for the $\rho_m$ variable using a 2-SBDF scheme with $\tau = 0.001$ . Blue signifies lowest values while red highest values. (a) Initial condition as random perturbation about $\rho_m = 1.0$ , (b) $\rho_m$ at time $t = 0.004$ , (c) $\rho_m$ at time $t = 2$ and (d) solution at final time $t = 4$ . . . . .	81
3.18	Solutions for the $\rho_a$ variable using 2-SBDF scheme with $\tau = 0.001$ . Blue signifies lowest values while red highest values. (a) Initial condition as random perturbation about $\rho_a = 1.0$ (b) $\rho_a$ at time $t = 0.004$ , (c) $\rho_a$ at time $t = 2$ (d) $\rho_a$ solution at final time $t = 4$ and (e) surface plot for the $\rho_a$ solution at final time $t = 4$ . . . . .	82
3.19	Graphical display of speed of the cell as the solution of the force balance equation for the cell. (a) Cell at initial stationary state, (b) cell speed at time $t = 2$ and (c) cell speed at final time $t = 4$ and (d) area of the evolving cell as a function of time showing increase in area of the cell with time. . . . .	83
3.20	Solutions for the $\rho_m$ variable using $\tau = 0.001$ . Blue signifies lowest values while red highest values. (a) Initial condition as random perturbation about $\rho_m = 1$ (b) $\rho_m$ at time $t = 0.004$ , (c) $\rho_m$ at time $t = 2$ and (d) solution at final time $t = 4$ . . . . .	84

3.21	Graphical displays of the numerical results of the F-actin concentration $\rho_a$ with increased total amount of actin $\rho_a^{tot} = 16$ using a 2-SBDF scheme with $\tau = 0.001$ and all other parameters held constant. Blue signifies lowest values while red highest values. (a) Initial condition as random perturbation about $\rho_a = 1.0$ (b) $\rho_a$ at time $t = 0.004$ , (c) $\rho_a$ at time $t = 2$ and (d) solution at final time $t = 4$ . . . . .	85
3.22	Speed of the cell as the solution of the force balance equation for the cell. (a) Cell at initial stationary state (b) cell speed at time $t = 0.004$ , (c) cell speed at time $t = 2$ , (d) cell speed at final time $t = 4$ and (e) area of the evolving cell as a function of time showing increase in area of the cell with time with increased total amount of actin $\rho_a^{tot} = 16$ . . . . .	86
3.23	Area of the evolving cell as a function of time. (a) area of the cell with contraction coefficient $\eta_1 = 0.2$ while all other parameters held constant and (b) decreasing area of the cell with a reduced total amount of actin $\rho_a^{tot} = 5$ while all other parameters held constant. . . . .	87
3.24	Cell shapes at different times showing the cell evolving with time. Red indicates highest values and blue lowest values. (a) cell shape at initial time, (b) cell shape at time $t = 2$ and (c) cell shape at time $t = 5$ . . . . .	100

# List of Abbreviations

ECM	Extra-Cellular Matrix
CAMS	Cell Adhesion Molecules
ATP	Adenosine Triphosphate
FAK	Focal Adhesion Kinase
F-actin	Actin filaments
G-actin	Globular actin (actin monomers)
RDEs	Reaction-Diffusion-Equations
PDEs	Partial Differential Equations
ODEs	Ordinary Differential Equations
IMEX	Implicit-Explicit scheme
SBDF	Semi-Implicit Backward Differentiation Formula
CNAB	Crank-Nicolson, Adams-Bashforth
CNLF	Crank-Nicolson, Leap-Frog
FEM	Finite Element Method
CG	Conjugate Gradient
PCG	Preconditioned Conjugate Gradient
GMRES	Generalised Minimal Residual method
sup	Supremum
lim	Limit
Re	Real part of
MATLAB	Matrix Laboratory

# Chapter 1

## Introduction

Cells are the smallest unit of life and widely vary in their shape, structure and function [Alberts et al. \(1995\)](#). They are complex in nature with different structural parts which work together to perform specific functions. Cells can exist on their own in the form of amoeba or exist as a group of cells as in multicellular organisms for example in plants and animals. Depending on their structural parts, cells are classified into several types. They are classified into two broad categories, namely prokaryotes and eukaryotes. Prokaryotic cells lack membrane bound organelles while eukaryotic cells are organised into complex structure with organelles enclosed within membranes [Alberts et al. \(1995\)](#). Cells perform their different functions by responding to their environment through deforming themselves and moving/migrating towards or away from stimuli. This is done by reorganising their internal structure. In this chapter, we give a biological overview of the cell, dynamics of cell migration, outline various models for cell migration that have been carried out and give an overview of numerical methods.

### 1.1 Cell structure and function

In this research, we will only concentrate on eukaryotic cells. In humans for example, there are more than 200 different types of cells each specialised for a specific function and are of order of tens of micrometres [Alberts et al. \(1995\)](#). All cells in an organism carry the same genome but as a result of differentiation, different cell types have different gene expression patterns [Alberts et al. \(1995\)](#); [Schwarz and Safran \(2013\)](#). Different cells come together to form tissues which work together to perform different functions. The formation of tissues and organs are brought about by division and movement of cells. This places cells in a crucial position in development of organisms.

#### 1.1.1 Structure of eukaryotic cell

For the biology of cells we make reference to [Alberts et al. \(1995\)](#). We will consider only eukaryotic cells. A eukaryotic cell consists of three broad and major parts, namely: plasma membrane, nucleus and cytoplasm.

## Plasma membrane

Plasma membrane is a phospholipid bilayer which encloses the contents of the cell and acts as a selective barrier [Alberts et al. \(1995\)](#). It also helps in communication of a cell with its surroundings or other cells. Communication between cells is mediated by extracellular signal molecules and the reception of these signals depend on the receptor protein that are found on the cell surface.

## Nucleus

Nucleus is a membrane bound organelle contained inside the cell and carries the genetic material of the cell. It occupies about ten percent of the total cell area [Alberts et al. \(1995\)](#). The bilayer membrane around the nucleus allows for selective transport of materials between the nucleus and other parts of the cell .

## Cytoplasm

Cytoplasm forms the rest of the cell and contains membrane-bound organelles. It is further subdivided into two parts: cytosol and cytoskeleton [Alberts et al. \(1995\)](#); [Pullarkat et al. \(2007\)](#). Cytosol is the solvent inside the cell where organelles are suspended. The cytoskeleton is a network of protein filaments and other associated proteins which render elasticity to the cell and control cell shape, locomotion and cell division [Pullarkat et al. \(2007\)](#). The cytoskeleton is made up of three filaments: actin filaments, microtubules and intermediate filaments which are distributed throughout the cell in an organised manner [Pullarkat et al. \(2007\)](#). The actin filaments form a meshwork structure below the plasma membrane, the microtubules originate from the centrosome close to the nucleus and extend all the way to the actin cortex while the intermediate filaments are concentrated around the nucleus and extend in lower number away from the nucleus [Pullarkat et al. \(2007\)](#); [Alberts et al. \(1995\)](#). Intermediate filaments provide the cell with mechanical strength, microtubules determine the positions of membrane enclosed organelles and direct intracellular transport while actin filaments (also referred to as F-actin) determine the shape of the cell's surface and are necessary for whole-cell movement [Pullarkat et al. \(2007\)](#); [Ananthakrishnan and Ehrlicher \(2007\)](#). A schematic representation of the cell is shown is Figure 1.1.

## Motor proteins

The filaments are linked together by different proteins. Motor proteins convert the energy of ATP hydrolysis into mechanical force that can either move organelles along the filaments or move the filaments themselves [Pullarkat et al. \(2007\)](#). These motor proteins generate forces when they interact with the filaments. Motor proteins are grouped into two categories depending on filaments along which they interact with, namely actin motors and microtubule motors [Alberts et al. \(1995\)](#); [Pullarkat et al. \(2007\)](#). Actin motors are made up of myosin family of proteins that carry out unidirectional movements along actin filaments thus generating contractile forces [Pullarkat et al. \(2007\)](#). The contractile forces play a crucial role in cell functioning. An example of myosin family is

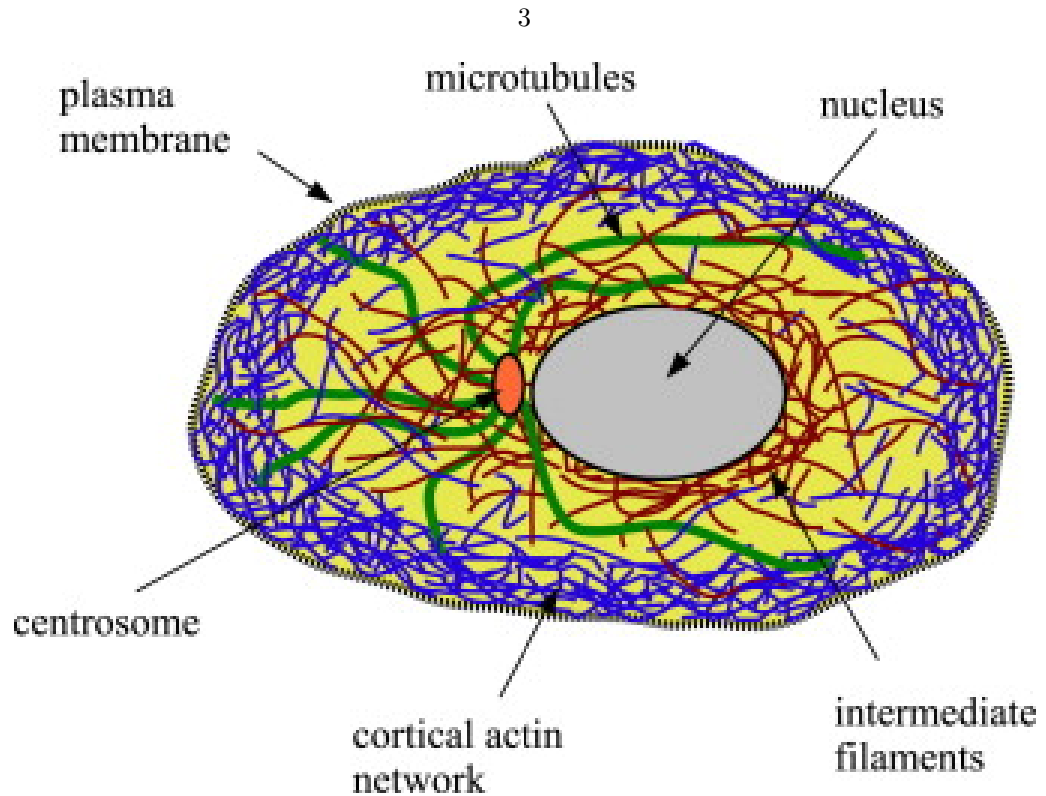


Figure 1.1: Schematic representation of the cell showing the components of the cell [Pullarkat et al. \(2007\)](#).

the myosin II which interact with actin filaments thereby generating active stresses. These stresses are necessary in providing cell with shape as well as controlling its movements. An example of microtubule motors is dyenin [Pullarkat et al. \(2007\)](#) which also play a role in cellular transport.

### Cross-linking proteins

Cross-linking proteins help in connecting the filaments and play an essential role in controlling assembly and disassembly of filaments [Ananthakrishnan and Ehrlicher \(2007\)](#) and in elasticity of actin filament network [Pullarkat et al. \(2007\)](#). The most common crosslinkers of actin filaments are  $\alpha$ -actinin and filamin [Pullarkat et al. \(2007\)](#).

### Regulatory proteins

Other types of proteins are the regulatory proteins which are also referred to as nucleation-promoting factors [Pollard \(2007\)](#); [Ananthakrishnan and Ehrlicher \(2007\)](#). They play an important role in controlling the assembly-disassembly dynamics of filaments and the activity of motor proteins [Pullarkat et al. \(2007\)](#).

### Actomyosin complexes

Actomyosin complexes are formed by the association of myosin motors with actin filaments [Pullarkat et al. \(2007\)](#). The myosin II self-assembles into short chains which can then act on neighbouring actin filaments to produce relative motion. Such complexes have a remarkable property of

generating active stresses within the actin network [Pullarkat et al. \(2007\)](#). Actomyosin complexes can be highly organised to form stress fibers and are therefore the main cytoskeletal component responsible for cell shape, generation of active stresses and locomotion. The actomyosin complexes therefore give the cell its viscoelastic properties [Pullarkat et al. \(2007\)](#).

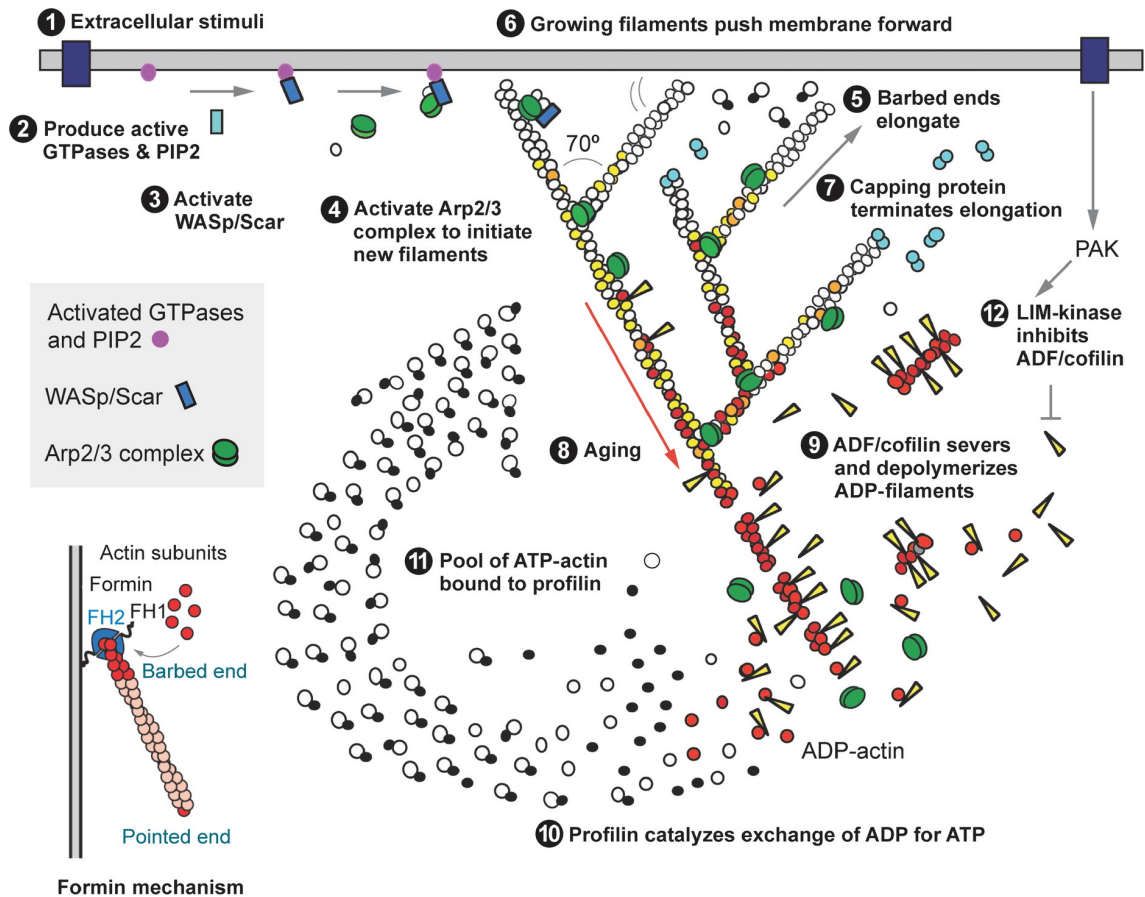
Actin occur in two forms: active form which is the F-actin and G-actin form (also called actin monomers) which is the inactive form and moves freely within the cytosol [Ananthakrishnan and Ehrlicher \(2007\)](#). Actin convert from the inactive state (G-actin) to the active state (F-actin) through a process called actin polymerisation and conversely from the active state to the inactive state through actin depolymerisation. The regulatory proteins are responsible for these actin polymerisation-depolymerisation processes [Pollard \(2007\)](#); [Ananthakrishnan and Ehrlicher \(2007\)](#). Actin polymerisation promoting proteins such as nucleator proteins helps in creating new actin filaments. On the other hand, actin depolymerisation factor for example cofilin is capable of binding to the actin filament thereby weakening it causing F-actin to break and form G-actin. It is largely believed that cytosol takes a passive role in transmission of stresses [Schwarz and Safran \(2013\)](#). On the outside, the cell is coupled to a multi-component gel-like network called the extracellular matrix which we abbreviate ECM.

### 1.1.2 Actin cytoskeleton and force generation

Of all the components of the cytoskeleton, F-actin and its corresponding motor proteins play the most crucial role in cell movement and therefore, we will focus on them. Actin occur in two forms: F-actin and G-actin forms. It converts from the inactive state (G-actin) to the active state (F-actin) through a process called actin polymerisation and conversely from the active state to the inactive state through actin depolymerisation [Ananthakrishnan and Ehrlicher \(2007\)](#). The regulatory proteins are responsible for these actin polymerisation-depolymerisation processes [Pollard \(2007\)](#); [Ananthakrishnan and Ehrlicher \(2007\)](#). Actin polymerisation promoting proteins such as nucleator proteins helps in creating new actin filaments. On the other hand, actin depolymerisation factor such as cofilin is capable of binding to the actin filament thereby weakening it causing F-actin to break and form G-actin monomers. Figure 1.2 summaries the dynamics of the actin polymerisation and depolymerisation. Actin filaments have two distinct ends: a plus end called barbed end which is a fast growing end and a minus end which is a slow growing end [Pollard and Borisy \(2003\)](#); [Pollard \(2007\)](#). Polymerisation happens when G-actin are added to the barbed end of F-actin [Pollard and Borisy \(2003\)](#). Actin filaments can assemble structures forming networks and bundles through interaction with motor proteins [Pellegrin and Mellor \(2007\)](#); [Pollard et al. \(2000\)](#). These structures produce cell protrusions called lamellipodia [Schwarz and Safran \(2013\)](#); [Pollard and Borisy \(2003\)](#); [Anderson and Cross \(2000\)](#). It is these activities of actin that lead to force generation. Actin cytoskeleton is therefore the main structure that contributes actively to force generation and therefore drives the cell movement [Schwarz and Safran \(2013\)](#); [Zhu et al. \(2000\)](#).

Actin filament is responsible for force generations that drive a cell forward through two major





**AR** Pollard TD. 2007.  
Annu. Rev. Biophys. Biomol. Struct. 36:451–77

Figure 1.2: Schematic representation of actin polymerisation and depolymerisation [Pollard and Borisy \(2003\)](#)

processes [Cuvelier et al. \(2007\)](#), namely: (i) rapid polymerisation of actin network at the cell periphery through the growth of lamellipodia [Schwarz and Safran \(2013\)](#); [Cavalcanti-Adam et al. \(2007\)](#); [Ridley \(2011\)](#); [Shemesh et al. \(2012\)](#). This leads to expansion of the plasma membrane and thus to the development of a contact area with the substrate and (ii) development of stress fibres and networks that are contractile due to the action of motor proteins that tend to slide actin filaments relative to each other [Schwarz and Safran \(2013\)](#); [Shemesh et al. \(2012\)](#). The most abundant motor proteins are the myosin II [Pellegrin and Mellor \(2007\)](#); [Burrige and Wittchen \(2013\)](#). These active forces from polymerisation of actin and contraction of stress fibres are eventually transmitted to the substrates through adhesion sites, providing the necessary forces required for cell propulsion [Ananthakrishnan and Ehrlicher \(2007\)](#); [Zhu et al. \(2000\)](#).

Plasma membrane plays an indirect role in force generation by controlling the polymerisation and the contraction of the stress fibres by triggering biochemical signals that regulate these processes [Schwarz and Safran \(2013\)](#); [Ridley \(2011\)](#). The plasma membrane also plays a role in the overall force balance in the cell since its tension and curvature elasticity provide counter-forces to actin-generated forces that tend to extend and deform the membrane [Schwarz and Safran \(2013\)](#).

There are counterforces exerted by the substrate on the cell. They originate from the substrate elasticity that resists deformation by the cellular forces. The substrate resistance can reorganise the cellular cytoskeleton and change the size of the adhesive regions [Ananthakrishnan and Ehrlicher \(2007\)](#); [Schwarz and Safran \(2013\)](#). This means that cellular structure and functions can be very sensitive to the elasticity and rigidity of the substrate.

### 1.1.3 Cell adhesion

Cell adhesion is the bonding of the cell to an object outside of the cell e.g when a cell binds to another cell or to the extracellular matrix (ECM) (we will call them substrates) [Liu et al. \(2007\)](#). Cellular adhesion is mediated by a wide variety of cell adhesion molecules (CAMs) also known as receptors and ligands which interact with each other [Liu et al. \(2007\)](#). The active stresses generated from actin polymerisation and contraction of myosin II motors are transmitted to the substrates via adhesion sites. These sites connect the actin cytoskeleton to adhesion receptors from the integrin family [Schwarz and Safran \(2013\)](#). The receptors then connect to the substrate or the ECM. The adhesion sites are mainly localised at the cell periphery because it is strongly coupled to the growth processes of the lamellipodium. Nascent adhesions are initiated close to the leading edge and then move towards the cell centre [Ananthakrishnan and Ehrlicher \(2007\)](#). This movement is mainly driven by the flow of actin away from the leading edge called retrograde flow [Schwarz and Safran \(2013\)](#); [Shemesh et al. \(2012\)](#); [Ananthakrishnan and Ehrlicher \(2007\)](#) due to the counterforces exerted on the polymerising actin by the membrane. As they move towards the cell centre, these small adhesions either mature into micrometer-sized focal adhesions or decay [Ananthakrishnan and Ehrlicher \(2007\)](#). The focal adhesions are responsible for attaching the cell to the substrate.

Transformation of the nascent adhesions into mature ones and the growth of the mature ones depends on the application of forces to them [Schwarz and Safran \(2013\)](#). The adhesions are stable only if sufficient force is exerted upon them and the larger the adhesion sites, the larger the force transmitted to the substrates [Schwarz and Safran \(2013\)](#). If filaments are sufficiently anchored to their surroundings, they can no longer slip back and this greatly helps the cell to propel forward. The force transmission to the substrates also largely depend on the substrates' stiffness [Schwarz and Safran \(2013\)](#). Cellular adhesion plays an important role in diverse biological and physiological processes including cell migration, wound healing, differentiation, thrombosis, tumor metastasis, arteriosclerosis and inflammation [Gupta \(2012\)](#); [Schwarz and Safran \(2013\)](#). The receptor-ligand bond lifetime has been shown to be finite even in the absence of an external force [Fournier et al. \(2010\)](#); [Liu et al. \(2007\)](#), that is, the individual receptor-ligand bonds will dissociate with or without an applied force [Qian et al. \(2009\)](#). This will allow the cell to propel forward.

Focal adhesions act not only as mechanical linkers that anchor the cell to the substrate, but also as prominent signalling centres that activate biochemical signalling molecules that diffuse into the cytoplasm and towards the nucleus [Vogel and Sheetz \(2009\)](#). Mature adhesions are known for signalling which upregulates both actin polymerisation and contractile myosin II motors. Many

signalling molecules responsible for cell migration and differentiation are localised to focal adhesions, most prominently the focal adhesion kinase (FAK) which is known to be important in many types of cancer [Mitra et al. \(2005\)](#). The molecular clutch comprising of talin, vinculin and paxillin connects the receptors to the actin cytoskeleton at the focal adhesion [Schwarz and Safran \(2013\)](#). These clutches determine the extent to which the cytoskeleton and the underlying substrate are linked and how they can interact by controlling the transmission of the cytoskeletal contractile forces to the substrate. When the clutch is engaged, there will be an effective transmission of the actomyosin contractile forces to the substrate leading to tight adhesion between the cell and the substrate. In contrast, when the clutch is not engaged, there is an ineffective force transmission to the substrate and thus adhesion is loosened [Schwarz and Safran \(2013\)](#); [Ananthakrishnan and Ehrlicher \(2007\)](#).

#### 1.1.4 Signaling pathways

During migration, the cell presents a front-rear polarity [Schaub et al. \(2007\)](#). This is brought about by different distributions of molecules in the front and rear parts of the cell. Depending on the type of the cell, some cells such as the Keratocytes, can polarise spontaneously in the absence of external cues [Mogilner and Edelstein-Keshet \(2002\)](#); [Mogilner and Keren \(2009\)](#). In other cells, polarity may be brought about by the extracellular environment. For example, the *Dictyostelium* cells are able to sense gradients of chemo-attractants and move towards their source [Buenemann et al. \(2010\)](#).

Multiple signaling pathways are initiated and organised at the cell surface to form a signalling network. This network is regulated during cell migration and help co-ordinate the processes of cell protrusion, adhesion, translocation of the cell body and retraction of the cell [Pullarkat et al. \(2007\)](#); [Buenemann et al. \(2010\)](#)

### Summary of steps for cell movement

Here, we summarise the processes of cell movement. Once a cell senses an external signal via the receptors that live on the surface, the leading edge begins to move in the direction of the signal by polymerising actin to form lamellipodia. This is called protrusion stage and is the first step of cell movement. Soon after the leading edge begins to protrude, adhesion molecules gathered in the extending region help attach the leading edge to the substrate. This is the adhesion stage and happens when actin bundles link the cytoskeleton to the substrate at the focal adhesions. The attachments prevent the leading edge from retracting. Also the cell de-adheres at the cell body and the rear of the cell. Finally the cell is propelled forward because of the contraction of stress fibres. This is the third stage of cell movement [Ananthakrishnan and Ehrlicher \(2007\)](#).

## 1.2 Models for cell migration

Cell migration refers to the trans-location of cells from one point to another. As we have seen in the section on the biology of the cell, the cytoskeleton plays a vital role in aiding cell migration and its mechanical strength through different processes. Cell migration has become one of the areas of much interest to many researchers from different fields including physics, computational sciences and mathematics. It is a fundamental process in many biological systems, for example wound healing, development of embryos, inflammation, cancer invasion, physiological process among others [Fidler \(2003\)](#); [Mogilner and Keren \(2009\)](#); [Pollard and Borisy \(2003\)](#); [Ridley et al. \(2003\)](#); [Lauffenburger and Horwitz \(1996\)](#).

For many centuries, experimental biology has occupied the minds of many researchers in the quest to understand the complexity of cell motility. In recent decades, mathematical and computational modelling has rapidly become an essential research technique that has greatly contributed to the understanding of the subject of cell motility [Mogilner \(2009\)](#). It is a fact that cell motility involves a large number of proteins that interact together in a complex way [Pullarkat et al. \(2007\)](#). Proposing an accurate model to account for the vast molecular interactions involved in cell motility is therefore a non-trivial activity. It is also largely known that the interaction of actin with its associated proteins is usually a major factor in the derivation of models for cell motility [Mogilner \(2009\)](#). Many models are built on the concept that cell motility is composed of the following stages: protrusion, adhesion and contraction [Ananthakrishnan and Ehrlicher \(2007\)](#). The cell pushes out the front, then it assembles tight adhesions to the surface at the leading edge and weakens such adhesions at the rear and finally the cell develops contractions that pull the weakly adherent rear towards the strongly adhered front [Mogilner \(2009\)](#).

There have been different strategies of modeling cell migration in the last several decades. The first modeling efforts were directed at quantifying actin treadmill and using thermodynamics to understand the nature and magnitude of the polymerisation force [Mogilner \(2009\)](#). These early works introduced fundamental ideas that are still used in developing complex models [Mogilner \(2009\)](#). In [Peskin et al. \(1993\)](#), 'Polymerisation Brownian ratchet' model was proposed to describe actin polymerisation as rigid mechanism which elongates by rectifying the Brownian motion of the membrane. According to this model, when the end of an actin filament comes into contact with a membrane, the membrane would diffuse away and therefore create a gap sufficient for monomers to be added [Peskin et al. \(1993\)](#); [Mogilner \(2009\)](#). These ratchet models used differential-difference equations [Peskin et al. \(1993\)](#); [Mogilner \(2009\)](#). Later [Mogilner and Oster \(1996\)](#) made an improvement to this model to consider the filaments as elastic springs whose behaviour is a function of the bending modulus of the filament and the angle it makes with a load at its tip. The thermal fluctuations of actin filaments displaces the actin filaments from the membrane and creates a gap for the elongation of the filament [Mogilner and Oster \(1996\)](#). This model was able to predict an optimal angle between the actin filament and the load for the effective force transmission. In [Mogilner and Oster \(2003\)](#), ratchet models underwent further development when

it was suggested that some of the actin filaments are attached to the surface they push. This model, the 'tethered ratchet' model, explains the mechanism by assuming that the filament attach to the surface transiently, dissociating fast and growing freely until getting capped and losing contact with the surface altogether [Mogilner and Oster \(2003\)](#). In [Marée et al. \(2006\)](#); [Satyanarayana and Baumgaertner \(2004\)](#); [Satulovsky et al. \(2008\)](#), actin polymerisation and depolymerisation were treated as stochastic processes.

[Mogilner and Edelstein-Keshet \(2002\)](#) developed a mathematical model that describes key details of actin dynamics in protrusion associated with cell motility. This model was based on the dendritic-nucleation hypothesis for lamellipodial protrusion in nonmuscle cells such as Keratocytes. An output of the model was a relationship between the protrusion velocity and the number of filament barbed ends pushing the membrane. They observed that this relationship has a local maximum: too many barbed ends deplete the available monomer pool and too few are insufficient to generate protrusive force. Their result suggested that to achieve rapid motility, some tuning of parameters affecting actin dynamics must be operating in the cell [Mogilner and Edelstein-Keshet \(2002\)](#).

Continuum models have also been developed to study cell motility. Examples are: a two-phase fluid model for cytosol and the actin network in [Alt and Dembo \(1999\)](#), a one-dimensional viscoelastic model of the cytoplasm and active stress generation in [Gracheva and Othmer \(2004\)](#), a one-dimensional model for the actin distribution and its reaction in [Mogilner et al. \(2001\)](#), a two-dimensional elastic continuum model in [Rubinstein et al. \(2005\)](#) and a cytomechanical model that couples a force balance mechanical equation for actin network to a reaction-diffusion equation for actin [Stephanou et al. \(2004\)](#). This cytomechanical model was later extended in [George \(2012\)](#) where they used a cartesian coordinate system.

Another modelling strategy is to use phase-field method [Biben and Misbah \(2003\)](#); [Biben et al. \(2005\)](#); [Liu et al. \(2007\)](#); [Zhang et al. \(2009\)](#); [Lowengrub et al. \(2009\)](#); [Shao \(2011\)](#); [Shao et al. \(2012\)](#). In [Shao et al. \(2010\)](#), Keratocyte cell was modelled as a two dimensional sheet with fixed area. The shape of the cell membrane was determined by the interactions of various forces including the surface tension, the bending forces and the pressure that constraints the cell area. In [Biben and Misbah \(2003\)](#); [Biben et al. \(2005\)](#), phase-field method has been used to simulate vesicles' deformation and tumbling while in [Liu et al. \(2007\)](#); [Zhang et al. \(2009\)](#), phase-field method was used to study the three dimension deformation of a vesicle membrane using an energetic framework. An extension of this model to multi-component vesicles was studied in [Lowengrub et al. \(2009\)](#). In [Shao et al. \(2012\)](#), a phase-field model for the Keratocyte cell that couples actin flow and adhesion mechanism during cell migration was presented. In the model, both myosin II contraction and actin polymerisation were treated as active stresses. The adhesion sites could switch between the gripping mode and the slipping mode and their dynamics were integrated with actin flow. They also included tension and bending forces at the membrane. The various forces involved in cell migration are then considered and translated into velocity which then evolve the cell shape. The model we consider is an extension of this model.

### 1.3 Numerical methods for models of cell migration

In many cases models for cell migration contain highly nonlinear reaction terms which make it impossible to obtain analytical solutions. Numerical methods are therefore a good choice in solving these models. Numerical methods for partial differential equations consist of two parts: a space discretisation to transform the system of partial differential equations into a system of ordinary differential equations and a time discretisation to transform the system of ordinary differential equations into a system of algebraic equations which can be solved using different techniques of linear algebra.

For the biochemical dynamics inside the cell, we will consider the well known model that was proposed by [Turing \(1952\)](#) that take the form

$$\frac{\partial \mathbf{u}}{\partial t} = \mathbf{D}\Delta \mathbf{u} + \mathbf{F}(\mathbf{u}), \quad (1.3.1)$$

where  $\mathbf{u}$  is the vector of chemical concentrations,  $\mathbf{F}$  represents the reaction kinetics and  $\mathbf{D}$  is the diagonal matrix of positive diffusion coefficients.

Space discretisation include finite differences [Morton and Mayers \(1998\)](#); [Mitchell and Griffiths \(1980\)](#), finite elements [Süli \(2007\)](#); [Reddy \(1993\)](#); [Houston et al. \(2002\)](#), boundary elements [Brebbia \(1980\)](#) among other methods. The moving grid finite element method was introduced in [Madzvamuse et al. \(2003, 2005\)](#) to study partial differential equations on complex evolving domains. It is a robust numerical method that has been applied to compute solutions of reaction-diffusion equations in continuously deforming domains [Madzvamuse et al. \(2003, 2005\)](#); [Madzvamuse \(2006\)](#). The finite element method is an efficient method that is able to deal with complex and irregular geometries and has been widely used for growing and deforming domains. In this thesis, it is sufficient to use finite differences and finite elements. For detailed theory on finite differences and finite elements, we make reference to [Morton and Mayers \(1998\)](#) and [Süli \(2007\)](#); [Reddy \(1993\)](#) respectively.

The space derivatives of (1.3.1) will be approximated by finite elements [Reddy \(1993\)](#) and finite differences [Morton and Mayers \(1998\)](#) to give a system of the form

$$\frac{\partial \mathbf{U}}{\partial t} = \mathbf{D}\Delta_h \mathbf{U} + \mathbf{F}(\mathbf{U}), \quad (1.3.2)$$

where  $\mathbf{D}\Delta_h \mathbf{U}$  and  $\mathbf{F}(\mathbf{U})$  arise from the diffusion and reaction terms respectively.

Several time discretisation have been used to obtain solutions for partial differential equations on both stationary and evolving domains. Fully explicit methods require very small time steps which result in computations that are expensive especially when it comes to multi-dimensions. [Ruuth \(1995\)](#) presented different IMPLICIT-EXPLICIT (hence IMEX) schemes. The key essence of these schemes is that an implicit scheme is applied to approximate the diffusive term and an explicit scheme is used to approximate the reaction kinetics. The IMEX schemes presented in [Ruuth \(1995\)](#); [Madzvamuse \(2006\)](#) include a first order semi-implicit backward differentiation formula (1-SBDF) which applies a backward differentiation formula to the diffusive term, Crank-Nicolson, Adams-Bashforth (CNAB) which applies Crank-Nicolson to the diffusive term and second

order Adams-Bashforth to the reaction terms, Crank-Nicolson Leap Frog (CNLF) which applies something similar to Crank-Nicolson to the diffusive term and a leap frog to the reaction terms, the second order semi-implicit backward differentiation formula (2-SBDF), which applies a second order formula to the diffusive term, the third order semi-implicit backward differentiation formula (3-SBDF) which applies a third order formula to the diffusive term and the first order backward Euler finite difference scheme (1-SBEM) which treats both the diffusive term and linear part of the reaction term implicitly and non-linear part of the reaction semi-implicitly. We note that the 1-SBDF, the 2-SBDF and the 3-SBDF schemes are known to give strong decay of high frequency error components while unfortunately the CNAB and CNLF schemes are known to give a weak damping of high frequency error components [Ruuth \(1995\)](#). From numerical experiments, the 2-SBDF is recommended as a good scheme to many two dimensional problems [Ruuth \(1995\)](#); [Madzvamuse \(2006\)](#). Recently, [Madzvamuse and Chung \(2014\)](#) used a fully implicit scheme to solve a system of bulk surface coupled reaction-diffusion equations. This scheme requires some special linearisation techniques as shown in [Madzvamuse and Chung \(2014\)](#). For this work, it is sufficient to apply the second order semi-implicit backward differentiation formula (2-SBDF). Applying the second order semi-implicit backward differentiation formula to (1.3.2) gives

$$\frac{3\mathbf{U}^{n+1} - 4\mathbf{U}^n + \mathbf{U}^{n-1}}{2\tau} = \mathbf{D}\Delta_h \mathbf{U}^{n+1} + 2\mathbf{F}(\mathbf{U}^n) - \mathbf{F}(\mathbf{U}^{n-1}), \quad (1.3.3)$$

where  $\tau = t^{n+1} - t^n$  is the time-step size [Ruuth \(1995\)](#). This method gives a strong decay of high frequency components [Ruuth \(1993\)](#).

## 1.4 Outline of the thesis

Hence, this thesis is structured as follows. In Chapter 2, we introduce some mathematical notations and formulae that will be used in the later parts of this thesis and formulate the viscous model for cell migration from conservation laws. This model includes reaction-advection-diffusion equations for actin and myosin II which represent the biochemical dynamics inside the cell and a force balance mechanical equation for the actin-myosin system. We also explore the phase-fields method and formulate the phase-field framework of the viscous model. This method introduces an auxiliary field, a free energy functional and a double well potential. The auxiliary function takes a value in one region and another value in another region and varies smoothly in the interface between the two regions. The double well potential has minima which describe the value of the auxiliary function in each region. In the context of the cell, one region would mean the interior of the cell and the other the exterior of the cell. The interface between the two regions of the cell will be the boundary of the cell. Furthermore, we non-dimensionalise both models and end up with non-dimensionalised viscous and phase-field models for cell migration.

These formulated model equations comprises of highly nonlinear partial differential equations with no closed form solutions. Therefore, in Chapter 3, we solve the models numerically. We start by considering the well studied Schnakenberg model and construct its numerical solvers. The



rationale behind this is that a lot of analytical results are known for the Schnakenberg model close to bifurcation points through the use of linear stability analysis. Once we have validated that our finite element solver is working for such complex nonlinear reaction-diffusion systems, we will then apply the solver to our original viscous model. It must be noted that reaction kinetics are a key component of the viscous model. The model equations we derive on growing domains assume that the domain deforms continuously. The most appropriate numerical technique to deal with domain growth is the moving finite element method [Reddy \(1993\)](#). Therefore, we outline the theory behind the finite element method and then apply it to the viscous model for cell migration and present numerical results for this model.

In Chapter 3, we also solve the phase-field model using finite differences for space discretisation. The original domain representing the evolving cell is defined inside a larger and regular domain. With this, we propose to use finite differences [Morton and Mayers \(1998\)](#) to discretise the new domain. We then solve the phase-field model using finite differences and present some preliminary results.

Finally, in Chapter 4, we summarise our work and outline possible extensions and future directions.



## Chapter 2

# Model formulation for cell migration

### 2.1 Introduction

In this chapter we introduce some mathematical concepts and formulae that will be used throughout this thesis and formulate model equations for cell migration that take into account the actin-myosin system as a machinery that drives cell movement. We will formulate a viscous model for cell motility that takes into account the bio-chemical nature of the cell and its mechanical structure as was discussed in Chapter 1. In particular, our model is based on two major principles: (i) F-actin and myosin II interact together inside the cell thereby generating forces that drive the cell and (ii) that the actin-myosin network is a viscous gel that exhibits the characteristics of viscous materials. The model consists of reaction-advection-diffusion equations for the F-actin and myosin II and a force balance mechanical equation for the actin-myosin system. Once formulated, we construct its corresponding phase-field framework to obtain a phase-field model for cell motility.

### 2.2 Mathematical preliminaries

#### Some basic notations

A two dimensional euclidean space is denoted as  $\mathbb{R}^2$  where

$$\mathbb{R}^2 = \{\mathbf{x} = (x, y) \mid x, y \in \mathbb{R}\}.$$

We will consider a subset of  $\mathbb{R}^2$  which is bounded and simply connected to be our domain. The domain representing a continuously deforming and moving cell at time  $t$  will be denoted by  $\Omega_t \subset \mathbb{R}^2$  with a corresponding boundary denoted by  $\partial\Omega_t$ , where  $t = (0, T]$  [Madzvamuse et al. \(2005\)](#). We also define  $d\Omega_t = dxdy$  while  $ds$  will be the element of arclength.

Let  $u(\mathbf{x}, t)$  be a  $C^1(\overline{\Omega}_t \times (0, T])$  scalar function,  $v(\mathbf{x}, t)$  be a  $C^2(\overline{\Omega}_t \times (0, T])$  scalar function,  $\mathbf{w}(\mathbf{x}, t) = (w_1, w_2)$  be a  $C^1(\overline{\Omega}_t \times (0, T])$  vector valued function and  $\mathbf{n} = (n_1, n_2)$  denote the unit

outward normal vector to  $\partial\Omega_t$ . We begin by defining the following operators as outlined in [Gilbarg and Trudinger \(2015\)](#):

The gradient of  $u$  is given by

$$\nabla u = \left( \frac{\partial u}{\partial x}, \frac{\partial u}{\partial y} \right).$$

The divergence of  $\mathbf{w}$  is given by

$$\nabla \cdot \mathbf{w} = \left( \frac{\partial w_1}{\partial x} + \frac{\partial w_2}{\partial y} \right).$$

The directional derivative of  $u$  in the direction  $\mathbf{n}$  is given by

$$\nabla u \cdot \mathbf{n} = \frac{\partial u}{\partial \mathbf{n}}.$$

The Laplacian of  $v$  is defined by

$$\Delta v = \nabla \cdot \nabla v = \left( \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right).$$

## Theorems

Let  $\Omega_t$  be a bounded and simply connected domain with  $C^1$  boundary  $\partial\Omega_t$ ,  $u(\mathbf{x}, t)$  and  $g(\mathbf{x}, t)$  be  $C^1(\overline{\Omega}_t \times (0, T])$  scalar functions,  $v(\mathbf{x}, t)$  be a  $C^2(\overline{\Omega}_t \times (0, T])$  scalar function,  $\mathbf{w}(\mathbf{x}, t) = (w_1, w_2)$  be a  $C^1(\overline{\Omega}_t \times (0, T])$  vector valued function and  $\mathbf{n} = (n_1, n_2)$  denote the unit outward normal vector to  $\partial\Omega_t$ . We then have the following theorems:

### Divergence theorem

The divergence theorem is given by

$$\int_{\Omega_t} \nabla \cdot \mathbf{w} \, d\Omega_t = \int_{\partial\Omega_t} \mathbf{w} \cdot \mathbf{n} \, ds, \quad (2.2.1)$$

where  $ds$  indicates the dimensional area element in  $\partial\Omega_t$  and  $\mathbf{n}$  is the outward unit normal to  $\partial\Omega_t$  [Gilbarg and Trudinger \(2015\)](#). In particular, if  $u$  is a  $C^2(\overline{\Omega}_t \times (0, T])$  function, we can take  $\mathbf{w} = \nabla u$  and have

$$\int_{\Omega_t} \Delta u \, d\Omega_t = \int_{\partial\Omega_t} \nabla u \cdot \mathbf{n} \, ds = \int_{\partial\Omega_t} \frac{\partial u}{\partial \mathbf{n}} \, ds. \quad (2.2.2)$$

### Gradient theorem

The gradient theorem of scalar function  $u$  is stated as [Reddy \(1993\)](#)

$$\int_{\Omega_t} \nabla u \, d\Omega_t = \int_{\partial\Omega_t} u \mathbf{n} \, ds, \quad (2.2.3)$$

where  $\mathbf{n} = (n_1, n_2)$  is the outward unit normal to  $\partial\Omega_t$ . Applying this formula to the product  $ug$  and expanding using product rule for differentiation gives

$$\int_{\Omega_t} g(\nabla u) \, d\Omega_t + \int_{\Omega_t} u(\nabla g) \, d\Omega_t = \int_{\partial\Omega_t} u g \mathbf{n} \, ds. \quad (2.2.4)$$

Therefore, in  $x$  and  $y$  directions, the gradient theorem becomes

$$\int_{\Omega_t} g \frac{\partial u}{\partial x} \, d\Omega_t + \int_{\Omega_t} u \frac{\partial g}{\partial x} \, d\Omega_t = \int_{\partial\Omega_t} u g n_1 \, ds, \quad (2.2.5)$$

$$\int_{\Omega_t} g \frac{\partial u}{\partial y} d\Omega_t + \int_{\Omega_t} u \frac{\partial g}{\partial y} d\Omega_t = \int_{\partial\Omega_t} u g n_2 ds, \quad (2.2.6)$$

respectively.

### Green's formula

Let  $\Omega_t$  be a bounded and simply connected domain with  $C^1$  boundary  $\partial\Omega_t$  for which the divergence theorem holds and let  $v$  and  $g$  be functions defined as above. We can select  $\mathbf{w} = g\nabla v$  in the divergence theorem (2.2.1), and have

$$\int_{\Omega_t} g \Delta v d\Omega_t + \int_{\Omega_t} \nabla v \cdot \nabla g d\Omega_t = \int_{\partial\Omega_t} g \frac{\partial v}{\partial \mathbf{n}} ds, \quad (2.2.7)$$

which is known as the Green's formula [Gilbarg and Trudinger \(2015\)](#); [Larsson and Thomée \(2003\)](#).

### Reynold's transport theorem

Let  $g(\mathbf{x}, t)$  be a scalar function in  $\Omega_t$  defined as above and  $\beta(\mathbf{x}, t)$  be a flow velocity field, then

$$\frac{d}{dt} \int_{\Omega_t} g d\Omega_t = \int_{\Omega_t} \left( \frac{Dg}{Dt} + g \nabla \cdot \beta \right) d\Omega_t, \quad (2.2.8)$$

where  $\frac{D}{Dt} = \frac{\partial}{\partial t} + \beta \cdot \nabla$  denotes the material derivative [Acheson \(1990\)](#); [Madzvamuse \(2000\)](#). We will use this theorem to derive the weak formulation of partial differential equations in Chapter 3.

### Fick's first law of diffusion

This law says that the flux,  $\mathbf{J}$ , of material which can be amount of chemical, cells and so on, is proportional to the gradient of the concentration of the material. Consider a material and let its concentration be given by a scalar function  $g(\mathbf{x}, t)$ . Then, diffusion flux,  $\mathbf{J}(\mathbf{x}, t)$  of the material is directly proportional to the gradient of the concentration  $g(\mathbf{x}, t)$  of the material. Mathematically, this law is

$$\mathbf{J} = -D \nabla g(\mathbf{x}, t), \quad (2.2.9)$$

where  $D$  is the diffusivity [Murray \(2002\)](#). The minus sign simply means that diffusion takes place from high to low concentration.

Having stated the theorems and mathematical concepts that we will require in model formulation, we now go ahead and derive the viscous model for cell migration.

## 2.3 Derivation of viscous model for cell migration

In many eukaryotic cells, migration is powered by the actin-myosin system and assisted by the adhesion of the cell to substrates. At the cell's leading edge, cross-linked actin filaments polymerise by adding actin monomers to their barbed ends while at the back of the cell myosin II binds to the bundled actin filaments and exerts contractile stress. The active stresses generated by the actin-myosin system are transmitted to the substrates through adhesion sites. The model system we consider is the Keratocyte cell which extends broad Lamellipodia and has a fan-like shape. The

viscous model for cell migration consists of three coupled partial differential equations representing three different components of cell movement. The first two equations are reaction-advection-diffusion equations describing concentrations of actin filaments and myosin II. These equations have diffusion terms for both actin filaments and myosin II, advection terms due to cell movement and reaction kinetics which represent polymerisation and depolymerisation. The third equation is a force balance mechanical equation for the actin-myosin network. For this equation, we will seek to compute velocity vector for the cell which will be the solution for this equation. The reaction-advection-diffusion equations are derived from mass conservation laws as outlined in [Murray \(2002\)](#); [Edelstein-Keshet \(1988\)](#) while the force balance mechanical equation is derived from continuum mechanics [Murray \(2002, 2001\)](#). The model consists of three variables: myosin II and actin filament densities and velocity of actin network. We will first derive reactions-advection-diffusion equations for F-actin and myosin II followed by force balance equation.

## Reaction-advection-diffusion equations for F-actin and myosin II

Reaction-diffusion equations have been used to model different biological phenomena to study pattern formation on fixed and growing domains [Murray \(2001, 2002\)](#); [Edelstein-Keshet \(1988\)](#); [Madzvamuse et al. \(2005\)](#). On fixed domains, reaction-diffusion equations generally take the form [Murray \(2002\)](#)

$$\frac{\partial \mathbf{u}}{\partial t} = \mathbf{D} \Delta \mathbf{u} + \mathbf{F}(\mathbf{u}), \quad (2.3.1)$$

with for example zero-flux boundary conditions. Here,  $\mathbf{u}$  is the vector of chemical concentrations,  $\mathbf{F}$  represents the reaction kinetics and  $\mathbf{D}$  is the diagonal matrix of positive diffusion coefficients. As we will be considering domain growth in two spatial dimensions, we will derive the equations on a two-dimensional evolving domain. We will be interested with the case of two species representing the dynamics of F-actin and myosin II.

Let  $\Omega_t \subset \mathbb{R}^2$  be a simply connected, bounded and continuously deforming domain representing the cell shape at time  $t \in (0, T]$ ,  $T > 0$  and  $\partial\Omega_t$  be the boundary of the cell with normal  $\mathbf{n}$  at a point  $\mathbf{x}(t)$ . At any point  $\mathbf{x}(t) = (x(t), y(t)) \in \Omega_t$ , let  $\rho_m = \rho_m(\mathbf{x}, t)$  be the myosin II concentration,  $\rho_a = \rho_a(\mathbf{x}(t), t)$  be the F-actin concentration in  $\Omega_t$  and  $\beta = \beta(\mathbf{x}, t)$  be the actin flow velocity. We consider a subset  $\mathbf{R}(t) \subseteq \Omega_t$  with boundary  $\partial\mathbf{R}(t)$ . Actin polymerisation occurs when actin monomers are added to the actin filaments at the leading edge of the cell [Pollard \(2007\)](#). Let this reaction be given by  $f(\rho_a(\mathbf{x}, t), \rho_a^{cyl}(\mathbf{x}, t))$  which depends on the actin filament concentration,  $\rho_a(\mathbf{x}, t)$  and also on the concentration of actin monomers in the cytosol  $\rho_a^{cyl}(\mathbf{x}, t)$ . We will give the precise reaction kinetics shortly. Myosin II is bound to the actin bundles at the rear end of the cell and generates contractile stress. The conservation equation states that the rate of change of the quantity of a material in  $\mathbf{R}(t) \subseteq \Omega_t$  is equal to the net flux of the material through the boundary  $\partial\mathbf{R}(t)$  plus the net creation of the material within the domain [Murray \(2002\)](#); [Edelstein-Keshet \(1988\)](#); [Madzvamuse \(2000\)](#). Applying the conservation equation to the dynamics of F-actin, we

have,

$$\frac{d}{dt} \int_{\mathbf{R}(t)} \rho_a(\mathbf{x}, t) d\mathbf{R}(t) = - \int_{\partial\mathbf{R}(t)} \mathbf{J} \cdot \mathbf{n} dS + \int_{\mathbf{R}(t)} f(\rho_a(\mathbf{x}, t), \rho_a^{cyl}(\mathbf{x}, t)) d\mathbf{R}(t), \quad (2.3.2)$$

where  $\mathbf{J} = \mathbf{J}(\mathbf{x}, t)$  is the flux of F-actin across the boundary, that is, the amount of F-actin crossing a unit area per unit time and  $\mathbf{n}$  is the normal vector to the boundary. Applying the divergence theorem (2.2.1) to the conservation equation above gives

$$\frac{d}{dt} \int_{\mathbf{R}(t)} \rho_a(\mathbf{x}, t) d\mathbf{R}(t) = \int_{\mathbf{R}(t)} (-\nabla \cdot \mathbf{J} + f(\rho_a(\mathbf{x}, t), \rho_a^{cyl}(\mathbf{x}, t))) d\mathbf{R}(t). \quad (2.3.3)$$

By using Reynold's transport theorem (2.2.8), we further get

$$\int_{\mathbf{R}(t)} \left( \frac{D\rho_a(\mathbf{x}, t)}{Dt} + \rho_a(\mathbf{x}, t) \nabla \cdot \boldsymbol{\beta}(\mathbf{x}, t) \right) d\mathbf{R}(t) = \int_{\mathbf{R}(t)} (-\nabla \cdot \mathbf{J} + f(\rho_a, \rho_a^{cyl})) d\mathbf{R}(t), \quad (2.3.4)$$

where  $\boldsymbol{\beta}(\mathbf{x}, t)$  is the actin flow velocity and  $\frac{D}{Dt} = \frac{\partial}{\partial t} + \boldsymbol{\beta} \cdot \nabla$  is the material derivative. Upon using the definition of material derivative to (2.3.4), we arrive at

$$\int_{\mathbf{R}(t)} \left( \frac{\partial \rho_a}{\partial t} + \boldsymbol{\beta} \cdot \nabla \rho_a + \rho_a \nabla \cdot \boldsymbol{\beta} \right) d\mathbf{R}(t) = \int_{\mathbf{R}(t)} (-\nabla \cdot \mathbf{J} + f(\rho_a, \rho_a^{cyl})) d\mathbf{R}(t), \quad (2.3.5)$$

where  $\mathbf{J} = \mathbf{J}(\mathbf{x}, t)$  and  $\rho_a = \rho_a(\mathbf{x}, t)$ . We next assume that actin flow from regions of high concentration to regions of low concentration and its flux  $\mathbf{J}$  is proportional to its concentration gradient. Using Fick's law of diffusion (2.2.9) and the product rule

$$\nabla \cdot (\rho_a \boldsymbol{\beta}) = \boldsymbol{\beta} \cdot \nabla \rho_a + \rho_a \nabla \cdot \boldsymbol{\beta},$$

we have

$$\int_{\mathbf{R}(t)} \left( \frac{\partial \rho_a}{\partial t} + \nabla \cdot (\rho_a \boldsymbol{\beta}) \right) d\mathbf{R}(t) = \int_{\mathbf{R}(t)} (D_a \Delta \rho_a + f(\rho_a, \rho_a^{cyl})) d\mathbf{R}(t), \quad (2.3.6)$$

where  $D_a$  is the F-actin diffusion coefficient. Finally, bringing the right hand side terms to the left gives

$$\int_{\mathbf{R}(t)} \left( \frac{\partial \rho_a}{\partial t} + \nabla \cdot (\rho_a \boldsymbol{\beta}) - D_a \Delta \rho_a - f(\rho_a, \rho_a^{cyl}) \right) d\mathbf{R}(t) = 0, \quad (2.3.7)$$

where  $\rho_a = \rho_a(\mathbf{x}, t)$ . Since equation (2.3.7) holds true for any arbitrary  $\mathbf{R}(t) \subseteq \boldsymbol{\Omega}_t$  for all time  $t$  and the integrands are continuous, we will have the following equation which represents the reaction-advection-diffusion equation for the variable  $\rho_a(\mathbf{x}, t)$

$$\frac{\partial \rho_a}{\partial t} + \nabla \cdot (\rho_a \boldsymbol{\beta}) - D_a \Delta \rho_a - f(\rho_a, \rho_a^{cyl}) = 0, \quad (2.3.8)$$

for all  $\mathbf{x} \in \boldsymbol{\Omega}_t$ .

A relevant boundary condition is the zero-flux boundary condition which specifies that there is no flow of actin into and out of the domain but it is contained inside the cell. We will consider this type of boundary condition. The variable  $\rho_a$  therefore satisfies the following reaction-advection-diffusion equation in  $\boldsymbol{\Omega}_t$  with zero flux boundary condition on  $\partial\boldsymbol{\Omega}_t$

$$\begin{cases} \frac{\partial \rho_a}{\partial t} + \nabla \cdot (\rho_a \boldsymbol{\beta}) = D_a \Delta \rho_a + f(\rho_a, \rho_a^{cyl}), & \mathbf{x} \in \boldsymbol{\Omega}_t, t \in (0, T], \\ \rho_a(\mathbf{x}, 0) = \rho_a^0(\mathbf{x}), & \mathbf{x} \in \boldsymbol{\Omega}_t, t = 0, \\ \frac{\partial \rho_a}{\partial \mathbf{n}} = 0, & \mathbf{x} \in \partial\boldsymbol{\Omega}_t, t \in (0, T], \end{cases} \quad (2.3.9)$$

with  $\rho_a = \rho_a(\mathbf{x}, t)$  and  $\beta = \beta(\mathbf{x}, t)$ . The second term on the left represents the advection term for the actin filaments, the first and second terms on the right are the diffusion and reaction terms respectively. The zero-flux boundary condition  $\frac{\partial \rho_a}{\partial \mathbf{n}} = 0$  specifies that actin filaments are confined into the domain and do not cross the boundary, neither are they increased by external sources. Actin exchanges between active membrane bound ( $\rho_a$ ) and inactive cytosolic ( $\rho_a^{cyl}$ ) states. We use a mass-conserved reaction kinetics  $f(\rho_a, \rho_a^{cyl})$  as in [Mori et al. \(2008\)](#); [Camley et al. \(2013\)](#) to describe the dynamics of  $\rho_a$  variable which is

$$f(\rho_a, \rho_a^{cyl}) = k_b \left( \frac{\rho_a^2}{K_a^2 + \rho_a^2} + k_a \right) \rho_a^{cyl} - k_c \rho_a, \quad (2.3.10)$$

with

$$\rho_a^{cyl} = \frac{\rho_a^{tot} - \int_{\Omega_t} \rho_a d\Omega_t}{\int_{\Omega_t} d\Omega_t}.$$

The first term on the right hand side of (2.3.10) represents the rate of polymerisation which is a function of  $\rho_a$  and  $\rho_a^{cyl}$  and the last term is a simple depolymerisation rate. For simplicity, we have assumed that there is a positive feedback from the active form  $\rho_a$  to its own production, i.e the conversion rate from  $\rho_a^{cyl}$  to  $\rho_a$  depends nonlinearly on  $\rho_a$  whereas the reverse conversion rate takes place at a constant basal rate. We represent the positive feedback by a Hill function with power 2. The parameters  $k_a$  and  $k_c$  are constant basal rates,  $k_b$  is the overall activation rate and  $K_a$  is a saturation parameter. Since the actin monomers are small molecules and can diffuse fast [Pollard \(2007\)](#), we assume that they are uniformly distributed throughout the cell. The parameter  $\rho_a^{tot}$  is the total amount of actin in the cell and  $\int_{\Omega_t} d\Omega_t$  is the area of the cell.

Myosin II is bound to a bundle of F-actin network at the rear of the cell and travels with actin flow [Pullarkat et al. \(2007\)](#). In this work, we assume that myosin II is advected together with the F-actin and diffuses inside the cell and is not created with time, that is, we will assume that myosin II is not added through reaction. Applying conservation law to myosin II concentration,  $\rho_m$ , we have

$$\frac{d}{dt} \int_{\mathbf{R}(t)} \rho_m(\mathbf{x}, t) d\mathbf{R}(t) = - \int_{\partial \mathbf{R}(t)} \mathbf{J} \cdot \mathbf{n} dS, \quad (2.3.11)$$

where  $\mathbf{J}$  is myosin II flux across the boundary and  $\mathbf{n}$  is the normal vector to the boundary. Applying the divergence theorem (2.2.1) to the conservation equation above gives

$$\frac{d}{dt} \int_{\mathbf{R}(t)} \rho_m(\mathbf{x}, t) d\mathbf{R}(t) = - \int_{\mathbf{R}(t)} \nabla \cdot \mathbf{J} d\mathbf{R}(t). \quad (2.3.12)$$

By using Reynold's transport theorem (2.2.8) and the material derivative  $\frac{D}{Dt} = \frac{\partial}{\partial t} + \beta \cdot \nabla$ , we get

$$\int_{\mathbf{R}(t)} \left( \frac{\partial \rho_m}{\partial t} + \beta \cdot \nabla \rho_m + \rho_m \nabla \cdot \beta \right) d\mathbf{R}(t) = - \int_{\mathbf{R}(t)} \nabla \cdot \mathbf{J} d\mathbf{R}(t). \quad (2.3.13)$$

We now assume that myosin II flows from regions of high concentration to regions of low concentration and its flux  $\mathbf{J}$  is proportional to its concentration gradient and similarly to the F-actin concentration and write it as  $\mathbf{J} = -D_m(\rho_a) \nabla \rho_m$ . This gives

$$\int_{\mathbf{R}(t)} \left( \frac{\partial \rho_m}{\partial t} + \nabla \cdot (\rho_m \beta) - \nabla \cdot D_m(\rho_a) \nabla \rho_m \right) d\mathbf{R}(t) = 0, \quad (2.3.14)$$

Since equation (2.3.14) holds true for any arbitrary  $\mathbf{R}(t) \subseteq \Omega_t$  for all time  $t$  and the integrands are continuous, we will have the following equation

$$\frac{\partial \rho_m}{\partial t} + \nabla \cdot (\rho_m \boldsymbol{\beta}) - \nabla \cdot D_m(\rho_a) \nabla \rho_m = 0, \quad (2.3.15)$$

for all  $\mathbf{x} \in \Omega_t$  which represents the dynamics of the variable  $\rho_m$ . We prescribe a zero-flux boundary condition for the variable  $\rho_m$  and have the dynamics of  $\rho_m$  as

$$\begin{cases} \frac{\partial \rho_m}{\partial t} + \nabla \cdot (\rho_m \boldsymbol{\beta}) = \nabla \cdot D_m(\rho_a) \nabla \rho_m, & \mathbf{x} \in \Omega_t, \ t \in (0, T], \\ \rho_m(\mathbf{x}, 0) = \rho_m^0(\mathbf{x}), & \mathbf{x} \in \Omega_t, \ t = 0, \\ \frac{\partial \rho_m}{\partial \mathbf{n}} = 0, & \mathbf{x} \in \partial \Omega_t, \ t \in (0, T] \end{cases} \quad (2.3.16)$$

where  $\rho_m(\mathbf{x}, t)$  is the myosin II concentration and  $\boldsymbol{\beta}(\mathbf{x}, t)$  is the actin flow velocity. The second term on the left represents advection term for myosin II and the term on the right is the diffusion term. We have prescribed a zero-flux boundary condition and set initial condition for  $\rho_m$ . The parameter  $D_m(\rho_a)$  is a positive diffusion coefficient depending on F-actin solution  $\rho_a$  such that it decreases with increasing F-actin concentration. We formulate it as was done in Camley et al. (2013), that is

$$D_m(\rho_a) = \frac{D_m^0}{1 + \frac{\rho_a}{K_D}}, \quad (2.3.17)$$

where  $D_m^0 \in \mathbb{R}^+$  is the maximum myosin II diffusion coefficient in the absence of  $\rho_a$  and  $K_D$  is the myosin II diffusion threshold.

For our model, actin and myosin II are the main source of active stresses which lead to protrusion and contraction of the cell. The reaction-advection-diffusion equations derived above will therefore be coupled to a force balance equation which we now present.

## Force balance mechanical equation for the actin-myosin system

We model the network of actin filaments in the cell as a viscous gel as was done in Murray (2001). The forces present in the cell are the viscous forces, polymerisation forces and contraction forces. It is assumed that within the cell, mechanisms adhere to Newtonian dynamics. The inertial effects are negligible compared to the viscous forces Lewis and Murray (1991). The viscous network of actin filaments interact with myosin II to generate contractile stress in the cell's cytoplasm. As the actin network continues to contract as a result of interaction with myosin II, actin filaments will become aligned into bundles and this will lead to more contraction because myosin II is favoured to work in the same direction Pullarkat et al. (2007). Similarly, at the cell's leading edge, cross-linked actin filaments polymerise by adding actin monomers to their barbed ends causing expansive stress at the cell periphery.

At any point  $\mathbf{x} = (x(t), y(t)) \in \Omega_t$ , let  $\boldsymbol{\beta}(\mathbf{x}, t)$  be the flow velocity of the elements of the actin network. When the actin network is at mechanical equilibrium, the sum of all forces acting at each point  $\mathbf{x} = (x(t), y(t)) \in \Omega_t$  will be zero Lewis and Murray (1991); Murray (2001); Rubinstein et al. (2009); Stephanou et al. (2004); Alt and Tranquillo (1995). Thus, in the quasi-steady state,

we have the following force balance equation in  $\Omega_t$

$$\begin{cases} \nabla \cdot (\boldsymbol{\sigma}_\nu(\mathbf{x}, t) + \boldsymbol{\sigma}_{myo}(\mathbf{x}, t) + \boldsymbol{\sigma}_{poly}(\mathbf{x}, t)) = \mathbf{0}, & \mathbf{x} \in \Omega_t, t \in (0, T], \\ \boldsymbol{\sigma}_\nu \cdot \mathbf{n} = \mathbf{0}, & \mathbf{x} \in \partial\Omega_t, t \in (0, T], \end{cases} \quad (2.3.18)$$

with stress free boundary condition. The terms  $\boldsymbol{\sigma}_\nu(\mathbf{x}, t)$ ,  $\boldsymbol{\sigma}_{myo}(\mathbf{x}, t)$  and  $\boldsymbol{\sigma}_{poly}(\mathbf{x}, t)$  are the viscous, myosin II driven contractile and actin generated expansive stress tensors respectively as described below.

### Myosin II driven contractile stress

Myosin II generates contractile stress and this stress is considered to have a magnitude proportional to the density of myosin II. Let  $\boldsymbol{\sigma}_{myo}(\mathbf{x}, t)$  denote this myosin II driven contractile stress,  $\rho_m(\mathbf{x}, t)$  denote myosin II density and  $\eta_m^0$  be the myosin II contraction coefficient. Then at each point  $\mathbf{x} = (x(t), y(t)) \in \Omega_t$ , we model this myosin II driven contractile stress as

$$\boldsymbol{\sigma}_{myo}(\mathbf{x}, t) = \eta_m^0 \rho_m(\mathbf{x}, t) \mathbf{I}, \quad \mathbf{x} \in \Omega_t, t \in (0, T], \quad \eta_m^0 \in \mathbb{R}^+, \quad (2.3.19)$$

where

$$\mathbf{I} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}. \quad (2.3.20)$$

This means that this contractile stress is isotropic and acts proportionately to the density of myosin II at each point inside the cell. Thus, the higher the myosin II density, the more the contraction.

### Viscous stress

Actomyosin network is treated as a viscous gel. As viscous materials, they deform and spread when external force is applied to them and do not return to their original state. We assume that the cell obeys newtonian dynamics in such a way that inertial terms is negligible compared to the viscous forces [Lewis and Murray \(1991\)](#). At any point  $\mathbf{x} = (x(t), y(t)) \in \Omega_t$ , let  $\boldsymbol{\beta}(\mathbf{x}, t)$  be the flow velocity of the elements of the actin network. The viscous stress is proportional to the rate of strain tensor and we model it as

$$\boldsymbol{\sigma}_\nu(\mathbf{x}, t) = \frac{\nu_0}{2} (\nabla \boldsymbol{\beta}(\mathbf{x}, t) + (\nabla \boldsymbol{\beta}(\mathbf{x}, t))^T), \quad \mathbf{x} \in \Omega_t, t \in (0, T], \quad \nu_0 \in \mathbb{R}^+, \quad (2.3.21)$$

where  $\frac{\nu_0}{2} (\nabla \boldsymbol{\beta}(\mathbf{x}, t) + (\nabla \boldsymbol{\beta}(\mathbf{x}, t))^T)$  is the strain rate tensor [Acheson \(1990\)](#); [Murray \(2001\)](#). We note that we have considered the cell actomyosin system only as a viscous gel unlike in [Stephanou et al. \(2004\)](#); [George \(2012\)](#) where they considered actin filaments as viscoelastic gel. A feature of viscoelastic materials is that they deform and spread when an external force is applied to them but they return back to their original state slowly.

### Actin generated expansive stress

Actin filaments are normally confined close to periphery of the cell and point outwards. They polymerise by adding actin monomers to their barbed ends hence apply protrusive force leading to



expansive stress at the cell periphery. This actin generated expansive stress is normally confined to the cell periphery because the actin filaments are oriented in such a way that their barbed ends, where actin monomers are added, are close to the cell membrane and point outwards [Pollard and Borisy \(2003\)](#). The polymerisation of actin generates expansive stress which is assumed to be normal to the membrane (no shear stress) and proportional to the cross linked actin filaments density. Let  $\sigma_{poly}(\mathbf{x}, t)$  be this expansive stress acting in the domain  $\Omega_t$  and  $\rho_a(\mathbf{x}, t)$  be the cross-linked actin filaments density. We assume that this expansive stress is confined to the periphery of the cell and is proportional to  $\rho_a(\mathbf{x}, t)$ . Then at each point  $\mathbf{x} = (x(t), y(t)) \in \Omega_t$ , we model the actin generated expansive stress as

$$\sigma_{poly}(\mathbf{x}, t) = -\eta_a^0 \rho_a(\mathbf{x}, t) \delta(l) \mathbf{I}, \quad \mathbf{x} \in \Omega_t, \quad t \in (0, T], \quad \eta_a^0 \in \mathbb{R}^+, \quad (2.3.22)$$

where  $\mathbf{I}$  is the identity tensor,  $\delta(l)$  labels the cell periphery and  $\eta_a^0$  is the F-actin protrusion coefficient. This polymerisation stress only acts in the cell periphery. In order to describe this stress, we will assume an initial domain of a disk and specify that  $\sigma_{poly}(\mathbf{x}, t)$  only acts in the region which is some distance  $l$  from the centre of the disk domain. With this we define the function  $\delta(l)$  to be of the form

$$\delta(l) = \begin{cases} 1 & \text{if the point } \mathbf{x} \text{ is such its distance from the origin of the disk is more than } l, \\ 0 & \text{otherwise.} \end{cases}$$

This means that far from the cell periphery, only the contractile stress acts on the cell and close to the cell periphery both the contractile and the actin generated polymerisation stresses act on the cell.

## The viscous model

We have formulated model equations that drives cell migration through the action of the actomyosin system. This model couples reaction-advection-diffusion of F-actin and myosin II to a force balance mechanical equation. We hereby give the summary of the model

$$\begin{cases} \frac{\partial \rho_m}{\partial t} + \nabla \cdot (\rho_m \beta) = \nabla \cdot D_m(\rho_a) \nabla \rho_m, & \mathbf{x} \in \Omega_t, \quad t \in (0, T], \\ \frac{\partial \rho_a}{\partial t} + \nabla \cdot (\rho_a \beta) = D_a \Delta \rho_a + f(\rho_a, \rho_a^{cyl}), & \mathbf{x} \in \Omega_t, \quad t \in (0, T], \\ \nabla \cdot (\sigma_\nu(\mathbf{x}, t) + \sigma_{myo}(\mathbf{x}, t) + \sigma_{poly}(\mathbf{x}, t)) = 0, & \mathbf{x} \in \Omega_t, \quad t \in (0, T], \\ \rho_m(\mathbf{x}, 0) = \rho_m^0(\mathbf{x}), \quad \rho_a(\mathbf{x}, 0) = \rho_a^0(\mathbf{x}), & \mathbf{x} \in \Omega_t, \quad t = 0, \\ \frac{\partial \rho_m}{\partial \mathbf{n}} = \frac{\partial \rho_a}{\partial \mathbf{n}} = 0, & \mathbf{x} \in \partial \Omega_t, \quad t \in (0, T], \\ \sigma_\nu \cdot \mathbf{n} = 0, & \mathbf{x} \in \partial \Omega_t, \quad t \in (0, T], \end{cases} \quad (2.3.23)$$

Parameters	Description	Value	Reference
$D_a$	Actin diffusion coefficient	$0.8\mu m^2/s$	Shao et al. (2012)
$D_m^0$	Max myosin II diffusion coefficient	$2\mu m^2/s$	Camley et al. (2013)
$\rho_m^0(\mathbf{x})$	Initial myosin II density	$0.2 - 0.4\mu m^{-2}$	Shao et al. (2012)
$\nu_0$	Shear viscosity coefficient for actin	$10^3 pNs/\mu m$	Camley et al. (2013)
$\eta_a^0$	F-actin protrusion coefficient	$560 pN\mu m^2$	Shao et al. (2012)
$\eta_m^0$	Myosin II contraction coefficient	$100 pN\mu m$	Shao et al. (2012)
$K_a$	Positive feedback threshold	$1\mu m^{-2}$	Camley et al. (2013)
$k_a$	Base polymerisation rate	0.01	Camley et al. (2013)
$k_c$	F-actin depolymerisation rate	$10s^{-1}$	Shao et al. (2012)
$k_b$	Overall polymerisation rate	$10s^{-1}$	Shao et al. (2012)
$K_D$	Myosin II diffusion threshold	$0.5\mu m^{-2}$	Camley et al. (2013)
$\rho_a^{tot}$	Total actin density	800	Shao et al. (2012)

Table 2.1: Dimensional parameters and their values.

where

$$\begin{cases}
 f(\rho_a, \rho_a^{cyl}) = k_b \left( \frac{\rho_a^2}{K_a^2 + \rho_a^2} + k_a \right) \rho_a^{cyl} - k_c \rho_a, \\
 \rho_a^{cyl} = \frac{\rho_a^{tot} - \int_{\Omega_t} \rho_a d\Omega_t}{\int_{\Omega_t} d\Omega_t}, \\
 \sigma_{myo}(\mathbf{x}, t) = \eta_m^0 \rho_m(\mathbf{x}, t) \mathbf{I}, \quad \eta_m^0 \in \mathbb{R}^+, \\
 \sigma_\nu(\mathbf{x}, t) = \frac{\nu_0}{2} (\nabla \beta(\mathbf{x}, t) + (\nabla \beta(\mathbf{x}, t))^T), \quad \nu_0 \in \mathbb{R}^+, \\
 \sigma_{poly}(\mathbf{x}, t) = -\eta_a^0 \rho_a(\mathbf{x}, t) \delta(l) \mathbf{I}, \quad \eta_a^0 \in \mathbb{R}^+,
 \end{cases} \quad (2.3.24)$$

where the variables  $\rho_a = \rho_a(\mathbf{x}, t)$  and  $\rho_m = \rho_m(\mathbf{x}, t)$  are the F-actin and myosin II concentrations respectively at a point  $\mathbf{x} = (x(t), y(t)) \in \Omega_t$ , the variable  $\beta(\mathbf{x}, t) = (\beta_1(\mathbf{x}, t), \beta_2(\mathbf{x}, t))$  is the flow velocity for the actin network,  $D_a$  is a positive diffusion coefficient for F-actin and  $D_m(\rho_a)$  is variable diffusion rate for myosin II. The function  $f(\rho_a(\mathbf{x}, t), \rho_a^{cyl}(t))$  is a reaction term which depends on the actin filament concentration,  $\rho_a(\mathbf{x}, t)$ , and also on the concentration of actin monomers in the cytosol  $\rho_a^{cyl}(t)$ . We let the initial domain  $\Omega_0$  to be a disk. The parameters for the dimensional model are listed in Table 2.1. Having formulated the viscous model, we now go ahead and non-dimensionalise the model. The non-dimensionalised parameters are displayed in Table 2.2.

## Non-dimensionalisation of the viscous model

We introduce non-dimensional rescaled variables as follows:

$$\begin{cases}
 \hat{\rho}_m = \frac{\rho_m}{K_a}, \quad \hat{\rho}_a = \frac{\rho_a}{K_a}, \quad \hat{\rho}_a^{cyl} = \frac{\rho_a^{cyl}}{K_a}, \quad \hat{\mathbf{x}} = \frac{\mathbf{x}}{R}, \quad \hat{\beta} = \frac{\beta}{k_b R}, \\
 \hat{t} = \frac{t}{\tau}, \quad \nabla = \frac{1}{R} \hat{\nabla}, \quad \Delta = \frac{1}{R^2} \hat{\Delta}, \quad \hat{\Omega}_t = \frac{\Omega_t}{R},
 \end{cases} \quad (2.3.25)$$

where  $\tau$  is the scaling factor for time and  $R$  is the scaling factor for length. We begin by non-dimensionalising the reaction-advection-diffusion equations. Substituting the new variables into

the reaction-diffusion equations for F-actin and myosin II gives

$$\frac{K_a}{\tau} \frac{\partial \hat{\rho}_m}{\partial \hat{t}} + k_b K_a \hat{\nabla} \cdot (\hat{\rho}_m \hat{\beta}) - \frac{K_a D_m^0}{R^2} \hat{\nabla} \cdot \left( \left( \frac{1}{1 + \frac{K_a \hat{\rho}_a}{K_D}} \right) \hat{\nabla} \hat{\rho}_m \right) = 0, \quad (2.3.26)$$

and

$$\frac{K_a}{\tau} \frac{\partial \hat{\rho}_a}{\partial \hat{t}} + k_b K_a \hat{\nabla} \cdot (\hat{\rho}_a \hat{\beta}) - \frac{D_a K_a}{R^2} \hat{\Delta} \hat{\rho}_a - k_b K_a \left( \frac{K_a^2 \hat{\rho}_a^2}{K_a^2 + K_a^2 \hat{\rho}_a^2} + k_a \right) \hat{\rho}_a^{cyl} + k_c K_a \hat{\rho}_a = 0, \quad (2.3.27)$$

Now divide (2.3.26) and (2.3.27) by  $\frac{K_a}{\tau}$  gives

$$\frac{\partial \hat{\rho}_m}{\partial \hat{t}} + k_b \tau \hat{\nabla} \cdot (\hat{\rho}_m \hat{\beta}) - \frac{\tau D_m^0}{R^2} \hat{\nabla} \cdot \left( \left( \frac{1}{1 + \frac{K_a \hat{\rho}_a}{K_D}} \right) \hat{\nabla} \hat{\rho}_m \right) = 0,$$

and

$$\frac{\partial \hat{\rho}_a}{\partial \hat{t}} + k_b \tau \hat{\nabla} \cdot (\hat{\rho}_a \hat{\beta}) - D_a \frac{\tau}{R^2} \hat{\Delta} \hat{\rho}_a - \tau k_b \left( \frac{K_a^2 \hat{\rho}_a^2}{K_a^2 + K_a^2 \hat{\rho}_a^2} + k_a \right) \hat{\rho}_a^{cyl} + k_c \tau \hat{\rho}_a = 0,$$

respectively. Choosing  $\frac{\tau D_m^0}{R^2} = 1$  gives

$$\frac{\partial \hat{\rho}_m}{\partial \hat{t}} + \frac{k_b R^2}{D_m^0} \hat{\nabla} \cdot (\hat{\rho}_m \hat{\beta}) - \hat{\nabla} \cdot \left( \left( \frac{1}{1 + \frac{K_a \hat{\rho}_a}{K_D}} \right) \hat{\nabla} \hat{\rho}_m \right) = 0,$$

and

$$\frac{\partial \hat{\rho}_a}{\partial \hat{t}} + \frac{k_b R^2}{D_m^0} \hat{\nabla} \cdot (\hat{\rho}_a \hat{\beta}) - \frac{D_a}{D_m^0} \hat{\Delta} \hat{\rho}_a - \frac{R^2 k_b}{D_m^0} \left( \frac{\hat{\rho}_a^2}{1 + \hat{\rho}_a^2} + k_a \right) \hat{\rho}_a^{cyl} + \frac{k_c R^2}{D_m^0} \hat{\rho}_a = 0.$$

Let  $d = \frac{D_a}{D_m^0}$ ,  $a = \frac{K_a}{K_D}$ ,  $k_3 = \frac{R^2 k_b}{D_m^0}$ ,  $k_4 = \frac{R^2 k_a k_b}{D_m^0}$ ,  $b = \frac{k_b R^2}{D_m^0}$  and  $e = \frac{R^2 k_c}{D_m^0}$  and have

$$\frac{\partial \hat{\rho}_m}{\partial \hat{t}} + b \hat{\nabla} \cdot (\hat{\rho}_m \hat{\beta}) - \hat{\nabla} \cdot \left( \left( \frac{1}{1 + a \hat{\rho}_a} \right) \hat{\nabla} \hat{\rho}_m \right) = 0,$$

and

$$\frac{\partial \hat{\rho}_a}{\partial \hat{t}} + b \hat{\nabla} \cdot (\hat{\rho}_a \hat{\beta}) - d \hat{\Delta} \hat{\rho}_a - \left( \frac{k_3 \hat{\rho}_a^2}{1 + \hat{\rho}_a^2} + k_4 \right) \hat{\rho}_a^{cyl} + e \hat{\rho}_a = 0.$$

Finally, for notational simplicity, we drop all hats to obtain the following non-dimensionalised system.

$$\begin{aligned} \frac{\partial \rho_m}{\partial t} + b \nabla \cdot (\rho_m \beta) &= \nabla \cdot \left( \left( \frac{1}{1 + a \rho_a} \right) \nabla \rho_m \right), \\ \frac{\partial \rho_a}{\partial t} + b \nabla \cdot (\rho_a \beta) &= d \Delta \rho_a + \left( \frac{k_3 \rho_a^2}{1 + \rho_a^2} + k_4 \right) \rho_a^{cyl} - e \rho_a, \end{aligned}$$

with

$$\rho_a^{cyl}(t) = \frac{\rho_a^{tot} - \int_{\Omega_t} \rho_a d\Omega_t}{\int_{\Omega_t} d\Omega_t}.$$

Similarly, the non-dimensionalised force balance mechanical equation is given by

$$\nabla \cdot (\sigma_\nu(\mathbf{x}, t) + \sigma_{myo}(\mathbf{x}, t) + \sigma_{poly}(\mathbf{x}, t)) = \mathbf{0},$$

where

$$\begin{cases} \sigma_{myo}(\mathbf{x}, t) = \eta_1 \rho_m(\mathbf{x}, t) \mathbf{I}, \quad \eta_1 \in \mathbb{R}^+, \\ \sigma_\nu(\mathbf{x}, t) = \nabla \beta(\mathbf{x}, t) + (\nabla \beta(\mathbf{x}, t))^T, \\ \sigma_{poly}(\mathbf{x}, t) = -\eta_2 \rho_a(\mathbf{x}, t) \delta(l) \mathbf{I}, \quad \eta_2 \in \mathbb{R}^+, \end{cases}$$

with

$$\eta_1 = \frac{2K_a \eta_m^0}{k_b \nu_0} \text{ and } \eta_2 = \frac{2K_a \eta_a^0}{k_b \nu_0}.$$

We choose  $R = 10.0 \mu m$  and have the non-dimensionalised parameters in Table 2.2.

Parameters	$a$	$k_3$	$k_4$	$b$	$d$	$e$	$\eta_1$	$\eta_2$	$\rho_a^{tot}$
Value	2.0	500.0	5.0	500.0	0.4	500.0	$\frac{1}{50}$	$\frac{56}{500}$	8.0

Table 2.2: Non-dimensional parameter values for the non-dimensionalised viscous model.

### The non-dimensionalised viscous model

The non-dimensionalised viscous model for cell migration is summarised below with parameters as given in Table 2.2.

$$\left\{ \begin{array}{l} \frac{\partial \rho_m}{\partial t} + b \nabla \cdot (\rho_m \beta) = \nabla \cdot \left( \left( \frac{1}{1+a\rho_a} \right) \nabla \rho_m \right), \quad \mathbf{x} \in \Omega_t, \quad t \in (0, T], \\ \frac{\partial \rho_a}{\partial t} + b \nabla \cdot (\rho_a \beta) = d \Delta \rho_a + \left( \frac{k_3 \rho_a^2}{1+\rho_a^2} + k_4 \right) \rho_a^{cyl} - e \rho_a, \quad \mathbf{x} \in \Omega_t, \quad t \in (0, T], \\ \nabla \cdot (\sigma_\nu(\mathbf{x}, t) + \sigma_{myo}(\mathbf{x}, t) + \sigma_{poly}(\mathbf{x}, t)) = \mathbf{0}, \quad \mathbf{x} \in \Omega_t, \quad t \in (0, T], \\ \rho_a(\mathbf{x}, 0) = \rho_a^0(\mathbf{x}), \quad \rho_m(\mathbf{x}, 0) = \rho_m^0(\mathbf{x}), \quad \mathbf{x} \in \Omega_t, \quad t = 0, \\ \sigma_\nu \cdot \mathbf{n} = \mathbf{0}, \quad \mathbf{x} \in \partial \Omega_t, \quad t \in (0, T], \\ \frac{\partial \rho_a}{\partial \mathbf{n}} = \frac{\partial \rho_m}{\partial \mathbf{n}} = 0, \quad \mathbf{x} \in \partial \Omega_t, \quad t \in (0, T], \end{array} \right. \quad (2.3.28)$$

with

$$\begin{aligned} \rho_a^{cyl}(t) &= \frac{\rho_a^{tot} - \int_{\Omega_t} \rho_a d\Omega_t}{\int_{\Omega_t} d\Omega_t}, \\ \sigma_{myo}(\mathbf{x}, t) &= \eta_1 \rho_m(\mathbf{x}, t) \mathbf{I}, \quad \eta_1 \in \mathbb{R}^+, \\ \sigma_\nu(\mathbf{x}, t) &= \nabla \beta(\mathbf{x}, t) + (\nabla \beta(\mathbf{x}, t))^T, \\ \sigma_{poly}(\mathbf{x}, t) &= -\eta_2 \rho_a(\mathbf{x}, t) \delta(l) \mathbf{I}, \quad \eta_2 \in \mathbb{R}^+, \end{aligned}$$

where  $\rho_a$ ,  $\rho_m$  and  $\beta$  are the dependent variables for this model. The initial domain is now the unit disk with radius  $r = 1$ . Since the polymerisation force is assumed to work only in the periphery of the cell, we let it act only in the region where the radius is  $r > 0.8$ . For subsequent time, we assume that there exists a family of bijective functions that map the point  $\boldsymbol{\eta} = (\eta_x, \eta_y)$  of the initial domain to point  $\mathbf{x} = (x, y)$  of the current domain  $\Omega_t$ . Consider the mapping  $l : \Omega_t \times (0, T] \rightarrow \mathbb{R}$  and its corresponding mapping  $\hat{l} : \Omega_0 \times (0, T] \rightarrow [0, 1]$  on the initial domain  $\Omega_0$  where  $\hat{l}(\boldsymbol{\eta}, t)$  represents the distance from the centre of  $\Omega_0$  and the point  $\boldsymbol{\eta}$  with  $l(\mathbf{x}(\boldsymbol{\eta}, t), t) = \hat{l}(\boldsymbol{\eta}, t)$ . To denote the region where the polymerisation force act, we define a delta function  $\delta(l)$  such that

$$\delta(l) = \begin{cases} 1 & \text{if the point } (\mathbf{x}, t) \text{ with } l(\mathbf{x}(\boldsymbol{\eta}, t), t) = \hat{l}(\boldsymbol{\eta}, t) \\ & \text{is such that the distance } \sqrt{\eta_x^2 + \eta_y^2} > 0.8 \text{ in the initial domain,} \\ 0 & \text{elsewhere.} \end{cases}$$

We have now obtained the non-dimensionalised viscous model for cell motility. In order to solve models such as the viscous model that we have derived, one would need to keep track of

the boundary. There are several methods that can be employed with no need to track the sharp evolving boundary. Phase-field method is one of these methods. We will explore this method in the next section and write the phase-field framework of the viscous model that we have already derived.

## 2.4 Phase-field model

### Introduction

Phase-field method has been used extensively to solve moving boundary problems. It has been applied to a number of problems such as crack propagation [Karma et al. \(2001\)](#), diffusional problems in complicated geometries [Teigen et al. \(2009\)](#); [Li et al. \(2009\)](#); [Kockelkoren et al. \(2003\)](#), solidification [Rappel \(2001\)](#), problems on mean-curvature flow [Du et al. \(2005\)](#); [Deckelnick et al. \(2005\)](#), surface PDEs [Dziuk and Elliott \(2013\)](#); [Rätz et al. \(2006\)](#), cell biology [Shao \(2011\)](#) among other moving boundary problems. This method avoids the tracking of the interface by introducing an auxiliary field that locates the interface and whose dynamics is coupled to other physical fields through an appropriate set of partial differential equations. Here, we will give mathematical concepts used in phase-field modeling and then derive the phase-field framework of the viscous model.

### Theory of phase-field modelling

To construct phase-field models, one needs to introduce an auxiliary field  $\phi(\mathbf{x}, t)$ , a free energy functional  $F[\phi]$  and a double well potential  $G(\phi)$ . The auxiliary field  $\phi(\mathbf{x}, t)$  (also known as the phase-field parameter) takes a value say  $\phi(\mathbf{x}, t) = 0$  in one region and another value  $\phi(\mathbf{x}, t) = 1$  in another region and varies smoothly between 0 and 1 in the interface between the two regions. The double well potential has minima which describes the value of  $\phi$  in each region. In the context of the cell, one region would mean the interior of the cell and the other the exterior of the cell. The interface between the two regions of the cell will be the boundary of the cell. In this case one needs to define a double well potential with two minima, such that the first minima would correspond to the value of the phase-field parameter  $\phi(\mathbf{x}, t)$  inside the cell and the second minima would correspond to the value of  $\phi(\mathbf{x}, t)$  outside the cell. The phase-field parameter will vary smoothly from one minima to the other across the region which represents the boundary of the cell.

On this note, let the inside of the cell be represented by  $\phi(\mathbf{x}, t) = 1$  and the exterior of the cell by  $\phi(\mathbf{x}, t) = 0$ . Let  $\phi(\mathbf{x}, t)$  vary smoothly from 0 to 1 across the boundary of the cell. We note that the cell membrane will be represented by  $\phi(\mathbf{x}, t) = \frac{1}{2}$ . To this end, we define a double well potential and free energy functional of the form

$$G(\phi) = 18\phi^2(1 - \phi)^2,$$

$$F[\phi] = \int_{\Omega} \left( \frac{\epsilon |\nabla \phi|^2}{2} + \frac{G(\phi)}{\epsilon} \right) d\mathbf{x} := \int_{\Omega} f(\mathbf{x}, \phi, \nabla \phi) d\mathbf{x}, \quad (2.4.1)$$

respectively where  $\epsilon$  is the width of the phase field [Du et al. \(2005\)](#).  $F[\phi]$  is assumed to have continuous first and second derivatives with respect to  $\phi$ . The double well potential has two minima  $\phi(\mathbf{x}, t) = 1$  and  $\phi(\mathbf{x}, t) = 0$  which distinguish the inside of the cell from the outside. The graph for the double well potential is shown in [Figure 2.1](#). We consider the simple case that the cell only possesses the interface energy (2.4.1) and assume that  $F[\phi]$  has continuous first and second derivatives with respect to  $\phi$ . The variational relation of the interface energy (2.4.1)

$$\epsilon \frac{\partial \phi}{\partial t} = - \frac{\delta F}{\delta \phi} \quad (2.4.2)$$

will give rise to the phase-field equation for the cell shape as shown below. Here,  $\frac{\delta F}{\delta \phi}$  is known as the variational derivative of the functional  $F[\phi]$ . In two-dimensions, the variational derivative is defined as [Gelfand et al. \(2000\)](#)

$$\frac{\delta F}{\delta \phi} = \frac{\partial f}{\partial \phi} - \sum_{i=1}^{i=2} \frac{\partial}{\partial x_i} \left( \frac{\partial f}{\partial \phi_{x_i}} \right). \quad (2.4.3)$$

Following (2.4.3), the variational relation of energy functional  $F[\phi]$  becomes

$$\begin{aligned} \epsilon \frac{\partial \phi}{\partial t} &= - \frac{\delta F[\phi]}{\delta \phi} \\ &= - \frac{G'(\phi)}{\epsilon} + \frac{\partial}{\partial x} \left( \frac{\partial}{\partial \phi_x} \left[ \frac{\epsilon}{2} \left( \left( \frac{\partial \phi}{\partial x} \right)^2 + \left( \frac{\partial \phi}{\partial y} \right)^2 \right) + \frac{G(\phi)}{\epsilon} \right] \right) \\ &\quad + \frac{\partial}{\partial y} \left( \frac{\partial}{\partial \phi_y} \left[ \frac{\epsilon}{2} \left( \left( \frac{\partial \phi}{\partial x} \right)^2 + \left( \frac{\partial \phi}{\partial y} \right)^2 \right) + \frac{G(\phi)}{\epsilon} \right] \right), \\ &= - \frac{G'(\phi)}{\epsilon} + \epsilon \frac{\partial}{\partial x} \left( \frac{\partial \phi}{\partial x} \right) + \epsilon \frac{\partial}{\partial y} \left( \frac{\partial \phi}{\partial y} \right) \\ &= - \frac{G'(\phi)}{\epsilon} + \epsilon \frac{\partial^2 \phi}{\partial x^2} + \epsilon \frac{\partial^2 \phi}{\partial y^2}, \\ &= \epsilon \Delta \phi - \frac{G'(\phi)}{\epsilon}, \end{aligned} \quad (2.4.4)$$

where  $G(\phi(\mathbf{x}, t))$  is a double well potential with minima at  $\phi(\mathbf{x}, t) = 1$  and  $\phi(\mathbf{x}, t) = 0$ . The variational relation is therefore given by

$$\epsilon \frac{\partial \phi}{\partial t} = \epsilon \Delta \phi - \frac{G'(\phi)}{\epsilon}. \quad (2.4.5)$$

However, in two dimension, the circle will shrink with time if we only consider the interface energy. To remedy this, we add a term to balance the interface energy. We follow the formulation in [Biben et al. \(2005\)](#); [Biben and Misbah \(2003\)](#) and have the following variational relation

$$\epsilon \frac{\partial \phi}{\partial t} = \epsilon \Delta \phi - \frac{G'(\phi)}{\epsilon} + c \epsilon |\nabla \phi|, \quad (2.4.6)$$

where  $c$  is the local interface curvature given by  $c = \nabla \cdot \frac{\nabla \phi}{|\nabla \phi|}$ .

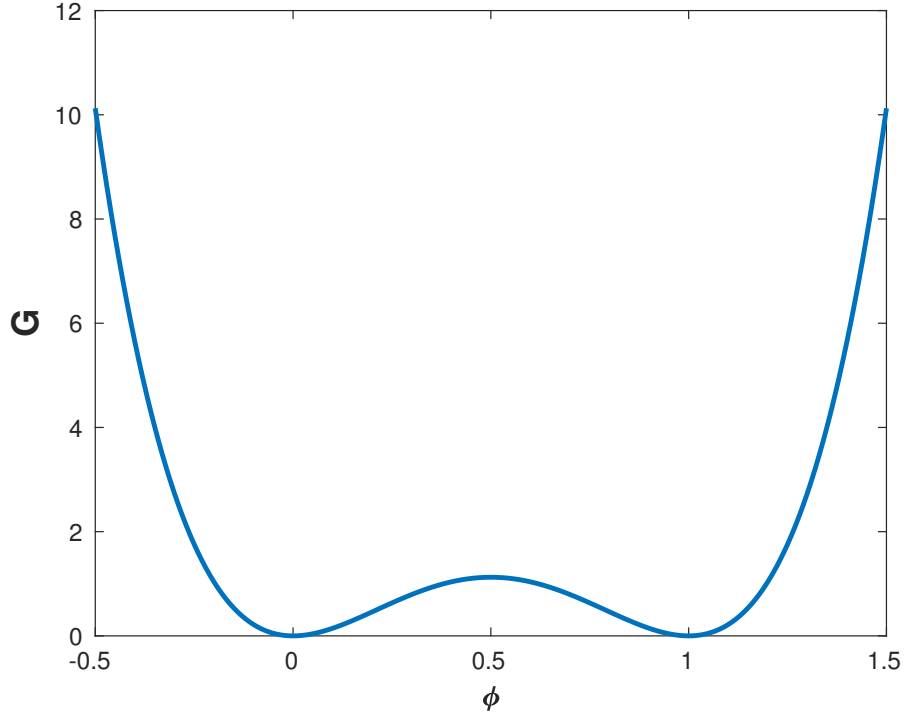


Figure 2.1: Double well potential for the phase-field.

The model equations for cell migration (2.3.23) can be formulated in a phase-field framework by adding an additional equation that will track the cell shape. We will summarise the phase-field framework for the equations governing cell migration.

### Phase-field framework for cell migration model

We formulate these equations in phase-field framework where cell shape is tracked by a phase-field  $\phi(\mathbf{x}, t)$  that takes value one inside the cell and zero outside. The phase-field parameter varies smoothly across the cell boundary, which is set by  $\phi(\mathbf{x}, t) = \frac{1}{2}$ . The equation for the cell shape is therefore given by

$$\frac{\partial \phi}{\partial t} + \nabla \cdot (\beta \phi) = \Gamma (\epsilon \Delta \phi - G'(\phi)/\epsilon + c\epsilon |\nabla \phi|), \quad \mathbf{x} \in \Omega, \quad t \in (0, T], \quad (2.4.7)$$

where  $\beta$  is the cytoskeletal velocity,  $\Gamma$  is a relaxation coefficient,  $c$  is the local interface curvature given by  $c = \nabla \cdot \frac{\nabla \phi}{|\nabla \phi|}$ ,  $\epsilon$  is a parameter for the interface width and  $G(\phi) = 18\phi^2(1 - \phi)^2$  is a double well potential with minima at  $\phi = 0$  and  $\phi = 1$  representing the outside and inside of the cell respectively. The reaction-advection-diffusion equations for myosin II and actin filaments in the phase-field framework are respectively given by

$$\frac{\partial(\phi \rho_m)}{\partial t} + \nabla \cdot (\phi \rho_m \beta) = \nabla \cdot (\phi D_m(\rho_a) \nabla \rho_m), \quad \mathbf{x} \in \Omega, \quad t \in (0, T], \quad (2.4.8)$$

$$\frac{\partial(\phi \rho_a)}{\partial t} + \nabla \cdot (\phi \rho_a \beta) = D_a \nabla \cdot (\phi \nabla \rho_a) + \phi f(\rho_a, \rho_a^{cyl}), \quad \mathbf{x} \in \Omega, \quad t \in (0, T], \quad (2.4.9)$$

where  $D_m(\rho_a)$  and  $f(\rho_a, \rho_a^{cyt})$  are the variable diffusion for myosin II and reaction kinetics for actin respectively and are given by

$$D_m(\rho_a) = \frac{D_m^0}{1 + \frac{\rho_a}{K_D}},$$

$$f(\rho_a, \rho_a^{cyt}) = k_b \left( \frac{\rho_a^2}{K_a^2 + \rho_a^2} + k_a \right) \rho_a^{cyt} - k_c \rho_a,$$

with

$$\rho_a^{cyt} = \frac{\rho_a^{tot} - \int_{\Omega_t} \phi \rho_a d\Omega_t}{\int_{\Omega_t} \phi d\Omega_t}.$$

The force balance equation is also written in the phase-field framework as

$$\nabla \cdot (\boldsymbol{\sigma}_\nu(\mathbf{x}, t) + \boldsymbol{\sigma}_{myo}(\mathbf{x}, t) + \boldsymbol{\sigma}_{poly}(\mathbf{x}, t)) = \mathbf{0}, \quad \mathbf{x} \in \Omega, \quad t \in (0, T], \quad (2.4.10)$$

where

$$\begin{cases} \boldsymbol{\sigma}_\nu(\mathbf{x}, t) = \frac{\nu_0}{2} \phi (\nabla \beta(\mathbf{x}, t) + (\nabla \beta(\mathbf{x}, t))^T), \quad \nu_0 \in \mathbb{R}^+ \\ \boldsymbol{\sigma}_{myo}(\mathbf{x}, t) = \eta_m^0 \phi \rho_m(\mathbf{x}, t) \mathbf{I}, \quad \eta_m^0 \in \mathbb{R}^+, \\ \boldsymbol{\sigma}_{poly}(\mathbf{x}, t) = -\eta_a^0 \phi \epsilon |\nabla \phi|^2 \rho_a(\mathbf{x}, t) \mathbf{I}, \quad \eta_a^0 \in \mathbb{R}^+, \end{cases} \quad (2.4.11)$$

where  $\epsilon |\nabla \phi|^2$  labels the cell periphery and  $\epsilon$  is the width of the interface.

### Non-dimensionalised phase-field model for cell migration

We carry out non-dimensionalisation as before by using the following rescaled variables

$$\hat{\rho}_m = \frac{\rho_m}{K_a}, \quad \hat{\rho}_a = \frac{\rho_a}{K_a}, \quad \hat{\rho}_a^{cyt} = \frac{\rho_a^{cyt}}{K_a}, \quad \hat{\mathbf{x}} = \frac{\mathbf{x}}{R}, \quad \hat{\beta} = \frac{\beta}{k_b R}, \quad \hat{t} = \frac{t}{\tau}, \quad \nabla = \frac{1}{R} \hat{\nabla}, \quad \Delta = \frac{1}{R^2} \hat{\Delta}, \quad (2.4.12)$$

where  $\tau$  is the scaling factor for time and  $R$  is the scaling factor for length. Substituting the new variables into the equation for the cell shape (2.4.7) and the reaction-diffusion equations for F-actin and myosin II gives

$$\frac{1}{\tau} \frac{\partial \phi}{\partial \hat{t}} + k_b \hat{\nabla} \cdot (\hat{\beta} \phi) = \Gamma \left( \frac{\epsilon}{R^2} \hat{\Delta} \phi - G'(\phi)/\epsilon + \frac{\epsilon}{R^2} \hat{c} |\hat{\nabla} \phi| \right), \quad (2.4.13)$$

$$\frac{K_a}{\tau} \frac{\partial (\phi \hat{\rho}_m)}{\partial \hat{t}} + k_b K_a \hat{\nabla} \cdot (\phi \hat{\rho}_m \hat{\beta}) - \frac{K_a D_m^0}{R^2} \hat{\nabla} \cdot \left( \left( \frac{\phi}{1 + \frac{K_a \hat{\rho}_a}{K_D}} \right) \hat{\nabla} \hat{\rho}_m \right) = 0, \quad (2.4.14)$$

and

$$\frac{K_a}{\tau} \frac{\partial (\phi \hat{\rho}_a)}{\partial \hat{t}} + k_b K_a \hat{\nabla} \cdot (\phi \hat{\rho}_a \hat{\beta}) - \frac{D_a K_a}{R^2} \hat{\nabla} \cdot (\phi \hat{\nabla} \hat{\rho}_a) = \phi \left( k_b K_a \left( \frac{K_a^2 \hat{\rho}_a^2}{K_a^2 + K_a^2 \hat{\rho}_a^2} + k_a \right) \hat{\rho}_a^{cyt} - k_c K_a \hat{\rho}_a \right), \quad (2.4.15)$$

Now divide (2.4.13) by  $\frac{1}{\tau}$  and (2.4.14) and (2.4.15) by  $\frac{K_a}{\tau}$  gives

$$\frac{\partial \phi}{\partial \hat{t}} + \tau k_b \hat{\nabla} \cdot (\hat{\beta} \phi) = \tau \Gamma \left( \frac{\epsilon}{R^2} \hat{\Delta} \phi - G'(\phi)/\epsilon + \frac{\epsilon}{R^2} \hat{c} |\hat{\nabla} \phi| \right),$$

$$\frac{\partial (\phi \hat{\rho}_m)}{\partial \hat{t}} + k_b \tau \hat{\nabla} \cdot (\phi \hat{\rho}_m \hat{\beta}) - \frac{\tau D_m^0}{R^2} \hat{\nabla} \cdot \left( \left( \frac{\phi}{1 + \frac{K_a \hat{\rho}_a}{K_D}} \right) \hat{\nabla} \hat{\rho}_m \right) = 0,$$

and

$$\frac{\partial (\phi \hat{\rho}_a)}{\partial \hat{t}} + k_b \tau \hat{\nabla} \cdot (\phi \hat{\rho}_a \hat{\beta}) - D_a \frac{\tau}{R^2} \hat{\nabla} \cdot (\phi \hat{\nabla} \hat{\rho}_a) = \phi \left( \tau k_b \left( \frac{K_a^2 \hat{\rho}_a^2}{K_a^2 + K_a^2 \hat{\rho}_a^2} + k_a \right) \hat{\rho}_a^{cyt} - k_c \tau \hat{\rho}_a \right),$$



respectively. Choosing  $\frac{\tau D_m^0}{R^2} = 1$  gives

$$\begin{aligned} \frac{\partial \phi}{\partial \hat{t}} + \frac{k_b R^2}{D_m^0} \hat{\nabla} \cdot (\hat{\beta} \phi) &= \frac{R^2}{D_m^0} \Gamma \left( \frac{\epsilon}{R^2} \hat{\Delta} \phi - G'(\phi)/\epsilon + \frac{\epsilon}{R^2} \hat{c} |\hat{\nabla} \phi| \right), \\ \frac{\partial(\phi \hat{\rho}_m)}{\partial \hat{t}} + \frac{k_b R^2}{D_m^0} \hat{\nabla} \cdot (\phi \hat{\rho}_m \hat{\beta}) - \hat{\nabla} \cdot \left( \left( \frac{\phi}{1 + \frac{K_a \hat{\rho}_a}{K_D}} \right) \hat{\nabla} \hat{\rho}_m \right) &= 0, \end{aligned}$$

and

$$\frac{\partial(\phi \hat{\rho}_a)}{\partial \hat{t}} + \frac{k_b R^2}{D_m^0} \hat{\nabla} \cdot (\phi \hat{\rho}_a \hat{\beta}) - \frac{D_a}{D_m^0} \hat{\nabla} \cdot (\phi \hat{\nabla} \hat{\rho}_a) = \phi \left( \frac{R^2 k_b}{D_m^0} \left( \frac{\hat{\rho}_a^2}{1 + \hat{\rho}_a^2} + k_a \right) \hat{\rho}_a^{cyl} - \frac{k_c R^2}{D_m^0} \hat{\rho}_a \right).$$

Let

$$\begin{cases} d = \frac{D_a}{D_m^0}, & a = \frac{K_a}{K_D}, \\ k_3 = \frac{R^2 k_b}{D_m^0}, & k_4 = \frac{R^2 k_a k_b}{D_m^0}, \\ b = \frac{k_b R^2}{D_m^0}, & e = \frac{R^2 k_c}{D_m^0}, \\ \gamma = \frac{\epsilon \Gamma}{D_m^0}, & \lambda = \frac{R^2 \Gamma}{\epsilon D_m^0}, \end{cases}$$

and have

$$\begin{aligned} \frac{\partial \phi}{\partial \hat{t}} + b \hat{\nabla} \cdot (\hat{\beta} \phi) &= \gamma \hat{\Delta} \phi - \lambda G'(\phi) + \gamma \hat{c} |\hat{\nabla} \phi|, \\ \frac{\partial(\phi \hat{\rho}_m)}{\partial \hat{t}} + b \hat{\nabla} \cdot (\phi \hat{\rho}_m \hat{\beta}) - \hat{\nabla} \cdot \left( \left( \frac{\phi}{1 + a \hat{\rho}_a} \right) \hat{\nabla} \hat{\rho}_m \right) &= 0, \end{aligned}$$

and

$$\frac{\partial(\phi \hat{\rho}_a)}{\partial \hat{t}} + b \hat{\nabla} \cdot (\phi \hat{\rho}_a \hat{\beta}) - d \hat{\nabla} \cdot (\phi \hat{\nabla} \hat{\rho}_a) = \phi \left( \left( \frac{k_3 \hat{\rho}_a^2}{1 + \hat{\rho}_a^2} + k_4 \right) \hat{\rho}_a^{cyl} - e \hat{\rho}_a \right).$$

Finally, for notational simplicity, we drop all hats to obtain the following non-dimensionalised system.

$$\begin{aligned} \frac{\partial \phi}{\partial t} + b \nabla \cdot (\phi \beta) &= \gamma \Delta \phi - \lambda G'(\phi) + \gamma c |\nabla \phi|, \\ \frac{\partial(\phi \rho_m)}{\partial t} + b \nabla \cdot (\phi \rho_m \beta) &= \nabla \cdot \left( \left( \frac{\phi}{1 + a \rho_a} \right) \nabla \rho_m \right), \\ \frac{\partial(\phi \rho_a)}{\partial t} + b \nabla \cdot (\phi \rho_a \beta) &= d \nabla \cdot (\phi \nabla \rho_a) + \phi \left( \left( \frac{k_3 \rho_a^2}{1 + \rho_a^2} + k_4 \right) \rho_a^{cyl} - e \rho_a \right), \end{aligned}$$

with

$$\rho_a^{cyl}(t) = \frac{\rho_a^{tot} - \int_{\Omega} \phi \rho_a d\Omega}{\int_{\Omega} \phi d\Omega}.$$

Similarly, the non-dimensionalised force balance mechanical equation is given by

$$\nabla \cdot (\sigma_{\nu}(\mathbf{x}, t) + \sigma_{myo}(\mathbf{x}, t) + \sigma_{poly}(\mathbf{x}, t)) = \mathbf{0}.$$

where

$$\begin{cases} \sigma_{myo}(\mathbf{x}, t) = \eta_3 \phi \rho_m(\mathbf{x}, t) \mathbf{I}, & \eta_3 \in \mathbb{R}^+, \\ \sigma_{\nu}(\mathbf{x}, t) = \phi (\nabla \beta(\mathbf{x}, t) + (\nabla \beta(\mathbf{x}, t))^T), \\ \sigma_{poly}(\mathbf{x}, t) = -\eta_4 \phi |\nabla \phi|^2 \rho_a(\mathbf{x}, t) \mathbf{I}, & \eta_4 \in \mathbb{R}^+, \end{cases}$$

with

$$\eta_3 = \frac{2K_a \eta_m^0}{k_b \nu_0} \text{ and } \eta_4 = \frac{2K_a \eta_a^0 \epsilon}{k_b \nu_0 R^2}.$$

Parameters	$a$	$k_3$	$k_4$	$b$	$d$	$e$	$\lambda$	$\gamma$	$\eta_3$	$\eta_4$	$\rho_a^{tot}$
Value	2.0	500.0	5.0	500.0	0.4	500.0	10	0.4	$\frac{1}{50}$	$\frac{112}{50,000}$	8.0

Table 2.3: Non-dimensionalised parameters for the phase-field model.

We end up with the following non-dimensionalised model in  $\Omega$ ,  $t \in (0, T]$  with non-dimensional parameters values in Table 2.3.

$$\begin{cases}
\frac{\partial \phi}{\partial t} + b \nabla \cdot (\phi \beta) = \gamma \Delta \phi - \lambda G'(\phi) + \gamma c |\nabla \phi|, \\
\frac{\partial(\phi \rho_m)}{\partial t} + b \nabla \cdot (\phi \rho_m \beta) = \nabla \cdot \left( \left( \frac{\phi}{1 + a \rho_a} \right) \nabla \rho_m \right), \\
\frac{\partial(\phi \rho_a)}{\partial t} + b \nabla \cdot (\phi \rho_a \beta) = d \nabla \cdot (\phi \nabla \rho_a) + \phi \left( \left( \frac{k_3 \rho_a^2}{1 + \rho_a^2} + k_4 \right) \rho_a^{cyl} - e \rho_a \right), \\
\nabla \cdot (\sigma_\nu(\mathbf{x}, t) + \sigma_{myo}(\mathbf{x}, t) + \sigma_{poly}(\mathbf{x}, t)) = \mathbf{0}, \\
\phi \rho_a(\mathbf{x}, 0) = \phi \rho_a^0(\mathbf{x}), \quad \phi \rho_m(\mathbf{x}, 0) = \phi \rho_m^0(\mathbf{x}), \quad \text{at } t = 0,
\end{cases} \quad (2.4.16)$$

where

$$\begin{cases}
\rho_a^{cyl}(t) = \frac{\rho_a^{tot} - \int_{\Omega} \phi \rho_a d\Omega}{\int_{\Omega} \phi d\Omega}, \\
c = \nabla \cdot \frac{\nabla \phi}{|\nabla \phi|}, \\
\sigma_\nu(\mathbf{x}, t) = \phi (\nabla \beta(\mathbf{x}, t) + (\nabla \beta(\mathbf{x}, t))^T), \\
\sigma_{myo}(\mathbf{x}, t) = \eta_3 \phi \rho_m(\mathbf{x}, t) \mathbf{I}, \quad \eta_3 \in \mathbb{R}^+, \\
\sigma_{poly}(\mathbf{x}, t) = -\eta_4 \phi |\nabla \phi|^2 \rho_a(\mathbf{x}, t) \mathbf{I}, \quad \eta_4 \in \mathbb{R}^+,
\end{cases} \quad (2.4.17)$$

with periodic boundary conditions.

## 2.5 Summary

In this chapter, we derived a viscous model for cell migration using conservation law. The model consists of three coupled equations, namely, reaction-advection-diffusion equations for F-actin and myosin II and a force balance equation for the actin-myosin system. The reaction-advection-diffusion equations assume that actin changes from its active state to inactive state and vice-versa and is advected inside the cell. Similarly, myosin II is bound to actin bundles and advected inside the cell. F-actin and myosin II are treated as the source of active stresses and their actions of polymerisation and contraction generate forces that drive the cell. Actin polymerisation takes place close to the periphery of the cell and results in expansion of the cell while contraction of myosin II leads to the cell being pulled inwards. We also considered the actin system as a viscous gel that exhibits the characteristics of a viscous material. At quasi-steady state, all the forces balance off and their sum is zero. The model has three variables, namely F-actin concentration, myosin II concentration and velocity of the actin-myosin network. Furthermore, we formulated the phase-field framework of the viscous model by introducing a phase-field parameter. Finally, we non-dimensionalised both models and ended up with non-dimensional parameters.

Now that we have derived the viscous and phase-field models for cell migration, we would like

to solve them. The model equations are highly nonlinear making analytical solutions difficult to get. We will therefore explore numerical solutions for these models in Chapter 3.

## Chapter 3

# Numerical methods for the viscous model for cell migration

### 3.1 Introduction

The aim of this chapter is to solve the viscous and phase-fields models (2.3.28) and (2.4.16) respectively. Our models, just as in many models for cell migration, contain highly nonlinear terms which make it impossible to obtain analytical solutions. In such cases where analytical solutions to mathematical problems are difficult to get, numerical methods serve as good choice to obtain the approximate solutions to the problems. Numerical methods for partial differential equations consist of two parts: a space discretisation to transform the system of partial differential equations into a system of ordinary differential equations and a time discretisation to transform the system of ordinary differential equations into a system of algebraic equations which can be solved using different techniques of linear algebra.

There are many numerical methods that have been developed and widely used to study different problems. When choosing a particular numerical method to solve a given problem, one may consider the ease in applying the method to the problem being solved, the ability of the method to generate accurate results in comparison to other numerical methods and the robustness of the numerical methods. Space discretisation include finite differences [Morton and Mayers \(1998\)](#); [Mitchell and Griffiths \(1980\)](#), finite elements [Süli \(2007\)](#); [Reddy \(1993\)](#), boundary elements [Hall \(1994\)](#); [Brebbia \(1980\)](#) among others. Finite differences have been used in different modelling aspects for example in solving problems in phase-fields [Camley et al. \(2013\)](#); [Crampin et al. \(1999\)](#); [Shao et al. \(2010, 2012\)](#). Their main advantages are that on fixed domains with simple geometries, they are simple to apply, easy to implement and can be easily parallelised [Morton and Mayers \(1998\)](#). However, their application to irregular domains which are continuously changing is usually non-trivial [Madzvamuse et al. \(2005\)](#). In such situations, finite elements are usually good choice. The moving grid finite element method was introduced to solve partial differential equations on complex evolving domains and has been applied to compute solutions of reaction-diffusion equations

in continuously deforming domains [Madzvamuse et al. \(2003, 2005\)](#); [Madzvamuse \(2006\)](#). In this thesis, it is sufficient to use finite differences and finite elements to discretise the domains.

Several time discretisation have been used to obtain solutions for partial differential equations on both stationary and evolving domains. Fully explicit methods require very small time steps which result in computations that are expensive especially when it comes to multi-dimensions while fully implicit schemes require some special linearisation techniques as shown in [Madzvamuse and Chung \(2014\)](#). [Ruuth \(1995\)](#) and [Madzvamuse \(2006\)](#) presented different IMplicit-EXplicit (hence IMEX) schemes. The key essence of these schemes is that an implicit scheme is applied to approximate the diffusive term and an explicit scheme is used to approximate the reaction kinetics. We will explore the second order semi-implicit backward differentiation formula (2-SBDF) scheme in this thesis. The 2-SBDF is an example of IMEX schemes. We begin by reviewing the theory of finite elements and then apply it to solve the viscous model.

## 3.2 The finite element method

### 3.2.1 Theory of the finite element method

We begin by summarising the main attributes of the finite element method for a partial differential equation of the form

$$\frac{\partial u}{\partial t} = \mathcal{L}u, \quad (3.2.1)$$

where  $\mathcal{L}$  is a differential operator containing  $u$  and its spatial derivatives and is in general nonlinear. For such problems, exact analytical solutions are not very trivial to get and therefore numerical techniques are usually sought in order to find the approximate solution to the problem. The basic idea behind the finite element method is to discretise the given continuous problem with infinitely many degrees of freedom to obtain a discrete system of equations with a finite number of degrees of freedom. The discretisation process begins with the reformulation of the given partial differential equation (3.2.1) to an equivalent weak form over some space  $V$  which we will define shortly. We multiply (3.2.1) by a test function  $v \in V$  and apply Greens theorem (2.2.7) to have the following problem: find  $u \in V$  such that

$$(v, \frac{\partial u}{\partial t}) = (v, \mathcal{L}u), \text{ for all } v \in V \quad (3.2.2)$$

where

$$(u, v) = \int_{\Omega_t} uv \, d\Omega_t, \quad (3.2.3)$$

is the  $L_2$ -inner product. To obtain a problem that can be solved numerically, we replace the space  $V$  by a finite-dimensional subspace  $V_h$  and seek both the approximate solution and test function  $v$  in this space. We get the following finite-dimensional problem: find  $u^h \in V_h$  such that

$$(v^h, \frac{\partial u^h}{\partial t}) = (v^h, \mathcal{L}u^h), \text{ for all } v^h \in V_h. \quad (3.2.4)$$

Next, we will subdivide the domain into sub-domains  $K_i$  and define a triangulation  $\mathcal{T}_h$  to be a finite union of the  $K_i$  and seek approximate  $u^h$  as a linear combination of a basis function of the

space  $V_h$ . In order to carry out the weak formulation and obtain the semi-discrete problem, we need to define the space with which to require the solution and also what sort of derivative to use. We define them below.

### Spaces and norms

Let  $\mathbb{N}$  denote the set of non-negative integers. An  $n$ -tuple

$$\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_n) \in \mathbb{N}^n$$

is called a multi-index and its length is given by  $|\boldsymbol{\alpha}| = \alpha_1 + \dots + \alpha_n$ .

#### $\mathbf{L}_p(\boldsymbol{\Omega}_t)$ space

Let  $\boldsymbol{\Omega}_t \subseteq \mathbb{R}^2$  be an open, bounded and continuously deforming domain with boundary  $\partial\boldsymbol{\Omega}_t$  and let  $p \geq 1$  be a real number. We define a space of integrable functions  $\mathbf{L}_p(\boldsymbol{\Omega}_t)$  on  $\boldsymbol{\Omega}_t$  by

$$\mathbf{L}_p(\boldsymbol{\Omega}_t) := \left\{ u(\mathbf{x}, t) \text{ a measurable function : } \int_{\boldsymbol{\Omega}_t} |u(\mathbf{x}, t)|^p d\boldsymbol{\Omega}_t < \infty \text{ for } \mathbf{x} \in \boldsymbol{\Omega}_t, t \in (0, T] \right\}.$$

The space  $\mathbf{L}_p(\boldsymbol{\Omega}_t)$  is equipped with the norm

$$\|u\|_{\mathbf{L}_p(\boldsymbol{\Omega}_t)} = \left( \int_{\boldsymbol{\Omega}_t} |u(\mathbf{x}, t)|^p d\boldsymbol{\Omega}_t \right)^{\frac{1}{p}}.$$

For  $p = \infty$ , the space  $\mathbf{L}_\infty(\boldsymbol{\Omega}_t)$  is equipped with the norm

$$\|u\|_{\mathbf{L}_\infty(\boldsymbol{\Omega}_t)} = \text{ess.sup}_{\mathbf{x}(t) \in \boldsymbol{\Omega}_t} |u(\mathbf{x}, t)|,$$

where  $\text{ess.sup}_{\mathbf{x}(t) \in \boldsymbol{\Omega}_t} |u(\mathbf{x}, t)|$  denotes the essential supremum of  $|u(\mathbf{x}, t)|$  [Süli \(2007\)](#).

#### Sobolev spaces

Having defined the  $\mathbf{L}_p(\boldsymbol{\Omega}_t)$  space, we now introduce a class of spaces which play an important role in the theory of the finite element method. Suppose that  $u(\mathbf{x}, t)$  is a locally integrable function defined on  $\boldsymbol{\Omega}_t$  and suppose also that there exists a function  $w_\alpha(\mathbf{x}, t)$  also locally integrable on  $\boldsymbol{\Omega}_t$  such that

$$\int_{\boldsymbol{\Omega}_t} w_\alpha(\mathbf{x}, t) \cdot v(\mathbf{x}, t) d\boldsymbol{\Omega}_t = (-1)^{|\alpha|} \int_{\boldsymbol{\Omega}_t} u(\mathbf{x}, t) \cdot \mathbf{D}^\alpha v(\mathbf{x}, t) d\boldsymbol{\Omega}_t \text{ for all } v(\mathbf{x}, t) \in \mathbf{C}_0^\infty,$$

then we say that  $w_\alpha(\mathbf{x}, t)$  is a weak derivative of the function  $u(\mathbf{x}, t)$  of order  $|\alpha| = \alpha_1 + \dots + \alpha_n$  and we write  $w_\alpha(\mathbf{x}, t) = \mathbf{D}^\alpha u(\mathbf{x}, t)$  [Süli \(2007\)](#). With this, we give an explicit definition of a Sobolev space. Let  $k$  be a non-negative integer and suppose that  $p \in [1, \infty]$  then

$$\mathbf{W}_p^k(\boldsymbol{\Omega}_t) = \{u \in \mathbf{L}_p(\boldsymbol{\Omega}_t) \mid \mathbf{D}^\alpha u(\mathbf{x}, t) \in \mathbf{L}_p(\boldsymbol{\Omega}_t), |\alpha| \leq k\}$$

is called a Sobolev space of order  $k$  [Süli \(2007\)](#). This space is equipped with the norm

$$\|u\|_{\mathbf{W}_p^k(\boldsymbol{\Omega}_t)} = \left( \sum_{|\alpha| \leq k} \|\mathbf{D}^\alpha u(\mathbf{x}, t)\|_{\mathbf{L}_p(\boldsymbol{\Omega}_t)}^p \right)^{\frac{1}{p}}, \text{ for } 1 \leq p < \infty,$$

and

$$\|u\|_{\mathbf{W}_\infty^k(\Omega_t)} = \sum_{|\alpha| \leq k} \|\mathbf{D}^\alpha u(\mathbf{x}, t)\|_{\mathbf{L}_\infty(\Omega_t)}, \quad \text{when } p = \infty.$$

An important case is when  $p = 2$  which is a Hilbert space  $\mathbf{W}_2^k(\Omega_t)$  commonly denoted by  $\mathbf{H}^k(\Omega_t)$  with inner product

$$(u, v)_{\mathbf{H}^k(\Omega_t)} := \sum_{|\alpha| \leq k} (\mathbf{D}^\alpha u(\mathbf{x}, t), \mathbf{D}^\alpha v(\mathbf{x}, t)).$$

Throughout this thesis, we will only be interested with the case  $k = 1$ , that is, the case  $\mathbf{H}^1(\Omega_t)$ . We let  $V = \mathbf{H}^1(\Omega_t)$  and define a conforming finite element space by  $V_h \subset V = \mathbf{H}^1(\Omega_t)$ . The solution  $u^h$  is known as the weak solution of the partial differential equation (3.2.1). A summary for finite element method is therefore to seek solutions of partial differential equations in the Sobolev space  $\mathbf{H}^1(\Omega_t)$ , discretise the domains using triangular or quadrilateral elements and seek solutions only in a finite dimensional space  $V_h$ .

To determine the values of the finite element approximate solutions to the semi-discrete problem (3.2.4), it remains to integrate the system of ordinary differential equations in time. There are two broad approaches to time-stepping, namely explicit and implicit methods. The simplest example of an explicit method is the forward Euler method. The forward Euler method for (3.2.1) is given by

$$\frac{u^{n+1} - u^n}{\tau} = \mathcal{L}u^n, \quad (3.2.5)$$

where  $\tau$  is the time-step size,  $u^n$  and  $u^{n+1}$  are the solutions at time  $t^n$  and  $t^{n+1}$  respectively and  $\tau = t^{n+1} - t^n$ . This method is conditionally stable Morton and Mayers (1998). For a moderately small mesh size  $h$ , this method requires very small time steps. This can be very restrictive. To avoid such restrictions, implicit methods are a natural choice Morton and Mayers (1998). There are two broad implicit methods, namely the backward Euler method and the trapezoidal rule. The backward Euler method for the partial differential equation (3.2.1) can be expressed as

$$\frac{u^{n+1} - u^n}{\tau} = \mathcal{L}u^{n+1}, \quad (3.2.6)$$

while the trapezoidal rule can be written in the general form as

$$\frac{u^{n+1} - u^n}{\tau} = \frac{1}{2} (\mathcal{L}u^n + \mathcal{L}u^{n+1}). \quad (3.2.7)$$

Both the backward Euler and trapezoidal rule are stable regardless of the time-step. For general linear diffusion problems, the implicit methods are unconditionally stable. The backward Euler method is a first order scheme in time while the trapezoidal rule is a second order scheme in time Morton and Mayers (1998).

Now that we have defined the space on which to seek solutions from, namely,  $\mathbf{H}^1(\Omega_t)$ , we are in a position to solve the viscous model (2.3.28). We take for illustrative purposes the well studied Schnakenberg model and then construct its numerical solvers. The rationale behind is that a lot of analytical results are known for the Schnakenberg model close to bifurcation points through the use of linear stability analysis. Once we have validated that our finite element solver is working for such complex nonlinear reaction-diffusion systems, we will then apply the solver to our original viscous model. It must be noted that reaction kinetics are a key component of the viscous model.

### 3.2.2 The Schnakenberg model

A lot of studies on reaction-diffusion equations on different domains with different reaction kinetics have been done and are well understood [Turing \(1952\)](#); [Murray \(2002\)](#); [Schnakenberg \(1979\)](#); [Madzvamuse \(2000, 2006\)](#); [Madzvamuse and Chung \(2014\)](#); [Chung \(2016\)](#). In particular, reaction-diffusion equations with Schnakenberg reaction kinetics have been considered and are well known on both stationary and evolving domains [Schnakenberg \(1979\)](#); [Murray \(2002\)](#); [Madzvamuse et al. \(2005\)](#); [Madzvamuse \(2000\)](#); [Chung \(2016\)](#). For illustrations, we only consider the Schnakenberg model on a stationary domain and construct its numerical solvers. We note that this solver can be extended to include the terms that come as a result of domain growth when one is considering model equations on evolving domains.

#### Model equations on a stationary domain

Let  $\Omega$  be a convex and stationary domain with Lipschitz boundary  $\partial\Omega$  and let  $I = (0, T]$  be some time interval. For illustrative purposes, we consider the well known Schnakenberg reaction kinetics and have the following model equations on  $\Omega$

$$\left\{ \begin{array}{l} \frac{\partial u}{\partial t} - \Delta u = \gamma(a - u + u^2v) := \gamma f(u, v), \quad \mathbf{x} \in \Omega, \quad t \in I, \\ \frac{\partial v}{\partial t} - d\Delta v = \gamma(b - u^2v) := \gamma g(u, v), \quad \mathbf{x} \in \Omega, \quad t \in I, \\ u(\mathbf{x}, 0) = u^0(\mathbf{x}), \quad \mathbf{x} \in \Omega, \quad t = 0, \\ \frac{\partial u}{\partial \mathbf{n}} = 0, \quad \mathbf{x} \in \partial\Omega, \quad t \in I, \\ v(\mathbf{x}, 0) = v^0(\mathbf{x}), \quad \mathbf{x} \in \Omega, \quad t = 0, \\ \frac{\partial v}{\partial \mathbf{n}} = 0, \quad \mathbf{x} \in \partial\Omega, \quad t \in I, \end{array} \right. \quad (3.2.8)$$

for concentrations  $u(\mathbf{x}, t)$  and  $v(\mathbf{x}, t)$  with  $a, b, d$  and  $\gamma$  being some real positive constants. Here,  $d$  measures the ratio of the relative diffusivity of the  $v$  to  $u$  variables while  $\gamma$  measures the strength of the reaction. For this system, we have chosen homogeneous Neumann boundary conditions on the entire boundary and initial conditions will be chosen to be small random perturbations about the steady state

$$(\hat{u}, \hat{v}) = \left( a + b, \frac{b}{(a + b)^2} \right). \quad (3.2.9)$$

#### Conditions for diffusion-driven instability

We derive briefly the conditions for diffusion-driven instability as showed by [Turing \(1952\)](#). In the absence of diffusion, (3.2.8) becomes

$$\frac{\partial u}{\partial t} = \gamma f(u, v) \quad \text{and} \quad \frac{\partial v}{\partial t} = \gamma g(u, v). \quad (3.2.10)$$

Define  $|\xi| \ll 1$  and  $|\eta| \ll 1$  such that  $(u, v) = (\hat{u} + \xi, \hat{v} + \eta)$  where  $(\hat{u}, \hat{v})$  is a homogeneous state of (3.2.10). Then linearising (3.2.10) about the steady state  $(\hat{u}, \hat{v})$  gives

$$\begin{pmatrix} \frac{\partial u}{\partial t} \\ \frac{\partial v}{\partial t} \end{pmatrix} = \begin{pmatrix} \frac{\partial f}{\partial \xi} \\ \frac{\partial g}{\partial \eta} \end{pmatrix} = \gamma \begin{pmatrix} f_u & f_v \\ g_u & g_v \end{pmatrix} \begin{pmatrix} \xi \\ \eta \end{pmatrix}, \quad (3.2.11)$$



where the derivatives of  $f$  and  $g$  are evaluated at the equilibrium point  $(\hat{u}, \hat{v})$ . Here, we note that in deriving (3.2.11), we have ignored the second order and higher order terms of  $\xi$  and  $\eta$ . By calculation, it can be shown that the equilibrium point  $(\hat{u}, \hat{v})$  is linearly stable provided

$$f_u + g_v < 0 \text{ and } f_u g_v - f_v g_u > 0, \quad (3.2.12)$$

where the derivatives are evaluated at the equilibrium point (3.2.9) Murray (2002). If one then allows spatial in-homogeneity, it is possible that the system evolves to an inhomogeneous steady state. This phenomenon is described as diffusion-driven instability or Turing instability Turing (1952); Murray (2002). We now consider the model in the presence of diffusion, i.e

$$\begin{cases} \frac{\partial u}{\partial t} - \Delta u = \gamma(a - u + u^2 v) := \gamma f(u, v), \\ \frac{\partial v}{\partial t} - d\Delta v = \gamma(b - u^2 v) := \gamma g(u, v), \end{cases} \quad (3.2.13)$$

and investigate under what conditions the equilibrium point  $(\hat{u}, \hat{v})$  becomes unstable when diffusion is added. Let  $\zeta = (\xi, \eta)^T$  and

$$\mathbf{J} = \begin{pmatrix} f_u & f_v \\ g_u & g_v \end{pmatrix}, \quad \mathbf{D} = \begin{pmatrix} 1 & 0 \\ 0 & d \end{pmatrix},$$

where the derivatives have been evaluated at the equilibrium point  $(\hat{u}, \hat{v})$ . Linearising (3.2.13) about the equilibrium point  $(\hat{u}, \hat{v})$ , we obtain

$$\frac{\partial \zeta}{\partial t} = \gamma \mathbf{J} \zeta + \mathbf{D} \Delta \zeta, \quad (3.2.14)$$

with homogeneous Neumann boundary conditions Murray (2001). The partial differential equation (3.2.14) can be solved by separation of variables to get

$$\zeta(\mathbf{x}, t) = \sum_k \mathbf{c}_k e^{\lambda t} \zeta_k(\mathbf{x}). \quad (3.2.15)$$

For each  $k$ ,  $\mathbf{c}_k$  represents the vector of Fourier coefficients and  $\zeta_k$  is the eigenfunction of the Laplacian Murray (2001), that is, it satisfies

$$\nabla^2 \zeta_k + k^2 \zeta_k = 0, \quad (3.2.16)$$

with zero flux boundary conditions

$$(\mathbf{n} \cdot \nabla) \zeta_k = 0. \quad (3.2.17)$$

Substituting (3.2.15) and (3.2.16) into (3.2.14), we obtain the following eigenvalue problem for each  $k$

$$(\lambda \mathbf{I} - \gamma \mathbf{J} + \mathbf{D} k^2) \mathbf{c}_k = \mathbf{0}, \quad (3.2.18)$$

where  $\mathbf{I}$  is the identity matrix. As we require nontrivial solutions for  $\mathbf{c}_k$ , the following must be satisfied Murray (2001)

$$|\lambda \mathbf{I} - \gamma \mathbf{J} + \mathbf{D} k^2| = 0. \quad (3.2.19)$$

It can be shown that  $\lambda = \lambda(k^2)$  satisfies the following dispersion relation

$$\lambda^2 + b(k^2) + c(k^2) = 0, \quad (3.2.20)$$

where

$$b(k^2) = k^2(1 + d) - \gamma(f_u + g_v), \quad (3.2.21)$$

$$c(k^2) = dk^4 - \gamma(df_u + g_v)k^2 + \gamma^2(f_u g_v - f_v g_u). \quad (3.2.22)$$

We note that the condition  $(f_u + g_v) < 0$  implies that  $b(k^2) > 0$  for all  $k^2$ . For diffusion-driven instability to occur, one of the roots of (3.2.20) must have  $Re\lambda(k^2) > 0$  for some  $k^2 > 0$ . We see that  $Re\lambda(k^2) > 0$  only when  $c(k^2) < 0$  for some  $k^2 > 0$ . Since the equation (3.2.22) is quadratic in terms of  $k^2$ , we can show easily that  $c(k^2) < 0$  for some  $k^2 > 0$ . This is possible if and only if

$$df_u + g_v > 0 \quad (3.2.23)$$

and

$$(df_u + g_v)^2 - 4d(f_u g_v - f_v g_u) > 0. \quad (3.2.24)$$

Therefore the conditions for diffusion-driven instability can be summarised as follows [Edelstein-Keshet \(1988\)](#); [Murray \(2001\)](#); [Madzvamuse \(2000\)](#): In the absence of diffusion

$$f_u + g_v < 0 \text{ and } f_u g_v - f_v g_u > 0, \quad (3.2.25)$$

and adding diffusion

$$df_u + g_v > 0 \text{ and } (df_u + g_v)^2 - 4d(f_u g_v - f_v g_u) > 0, \quad (3.2.26)$$

where the derivatives have been evaluated at the equilibrium point. The choices  $a = 0.1$ ,  $b = 0.9$ ,  $d = 10$  and  $\gamma = 29$  will lead to diffusion-driven instability [Madzvamuse \(2000\)](#). For linear stability analysis of this model, we make reference to [Murray \(2002\)](#); [Madzvamuse \(2000\)](#).

## Numerical methods

Due to the non-linearities in the Schnakenberg model (3.2.8), its analytical solution is not readily available and therefore we seek its numerical solutions using the finite element method. The finite element method is based on the idea of dividing a domain into sub-domains such that the whole domain is approximated as a collection of these sub-domains. These sub-domains are known as elements while their vertices are called nodes. The model equations are then described on each element and solution is approximated over the elements by polynomials in terms of values at the nodes and then assembled at the nodes to form an approximate system of equations for the entire domain. For time dependent problems, this formulation renders the problem into a system of ordinary differential equations and therefore a time discretisation is required to transform the system of ordinary differential equations into a system of algebraic equations which when solved gives values of the unknown quantities at the nodes. This temporal discretisation is often achieved using finite differences. The finite element method therefore involves the following steps: derivation of weak formulation of the partial differential equations, the finite element spatial discretisation to obtain a system of semi-discrete equations and a temporal discretisation to obtain fully discrete equations.

### Weak formulation

To derive the weak formulation of the reaction-diffusion equation (3.2.8), we multiply each of the equations by a test function say  $\psi(\mathbf{x}, t) \in H^1(\Omega)$  and integrate over  $\Omega$  as follows

$$\int_{\Omega} \psi \frac{\partial u}{\partial t} d\Omega - \int_{\Omega} \psi \Delta u d\Omega = \gamma \int_{\Omega} \psi (a - u + u^2 v) d\Omega, \quad (3.2.27)$$

and

$$\int_{\Omega} \psi \frac{\partial v}{\partial t} d\Omega - d \int_{\Omega} \psi \Delta v d\Omega = \gamma \int_{\Omega} \psi (b - u^2 v) d\Omega. \quad (3.2.28)$$

Now we apply Green's formula to the above equations and use  $\frac{\partial u}{\partial \mathbf{n}} = \frac{\partial v}{\partial \mathbf{n}} = 0$  on  $\partial\Omega$  to arrive at the following weak formulation: find  $u(\mathbf{x}, t), v(\mathbf{x}, t) \in H^1(\Omega)$ ,  $t \in (0, T]$  such that

$$\begin{cases} \int_{\Omega} \psi \frac{\partial u}{\partial t} d\Omega + \int_{\Omega} \nabla \psi \cdot \nabla u d\Omega = \gamma \int_{\Omega} \psi (a - u + u^2 v) d\Omega, \\ \int_{\Omega} \psi \frac{\partial v}{\partial t} d\Omega + d \int_{\Omega} \nabla \psi \cdot \nabla v d\Omega = \gamma \int_{\Omega} \psi (b - u^2 v) d\Omega, \end{cases} \quad (3.2.29)$$

for all  $\psi(\mathbf{x}, t) \in H^1(\Omega)$ .

### The finite element spatial discretisation

Let  $\Omega_h$  be the computational domain which is a polyhedral approximation to  $\Omega$ . We define  $T_h$  to be a triangulation of  $\Omega_h$  made up of non-degenerate rectangular elements  $K_i$  such that  $T_h = \bigcup_i K_i$ . We call each  $K_i$  an element of the mesh  $T_h$  where  $h$  is the diameter of the largest element. For the mesh  $T_h$ , we require that it is made up of a finite number of elements and the elements must intersect along a complete edge, or at a vertex or not at all. We carry out the space discretisation using quadrilateral elements and seek a piece-wise linear approximation of the solution. To this end, we define the finite element space  $V_h \subset \mathbf{H}^1(\Omega_t)$  by

$$V_h = \{v_h \in C^0(\Omega) : v_h|_K \text{ is linear}\} \quad (3.2.30)$$

We will seek solutions of the Schnakenberg model in this space. The discretised version of the weak forms of the reaction-diffusion equations above therefore reads: find  $u^h(\mathbf{x}, t), v^h(\mathbf{x}, t) \in V_h$  such that

$$\begin{cases} \int_{\Omega_h} \psi^h \frac{\partial u^h}{\partial t} d\Omega_h + \int_{\Omega_h} \nabla \psi^h \cdot \nabla u^h d\Omega_h = \gamma \int_{\Omega_h} \psi^h (a - u^h + (u^h)^2 v^h) d\Omega_h, \\ \int_{\Omega_h} \psi^h \frac{\partial v^h}{\partial t} d\Omega_h + d \int_{\Omega_h} \nabla \psi^h \cdot \nabla v^h d\Omega_h = \gamma \int_{\Omega_h} \psi^h (b - (u^h)^2 v^h) d\Omega_h, \end{cases} \quad (3.2.31)$$

for all  $\psi^h \in V_h$ . Since  $V_h$  is a linear space, it must have a basis. We introduce the basis function

$$\phi_i(\mathbf{x}) \in V_h, \quad i = 1, 2, \dots, N_h \quad (3.2.32)$$

such that

$$\phi_i(p_j) = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{if } i \neq j, \end{cases} \quad (3.2.33)$$

where  $p_j$  is the  $j$ th nodal point of the mesh. The basis function  $\phi_i(\mathbf{x})$  is called piece-wise linear finite element nodal basis functions. We seek to find the finite element numerical approximations

$u^h(\mathbf{x}, t), v^h(\mathbf{x}, t) \in V_h$  expressed as linear combinations of the linear nodal basis functions  $\phi_i(\mathbf{x})$  as follows

$$u^h(\mathbf{x}, t) = \sum_{j=1}^{N_h} U_j(t) \phi_j(\mathbf{x}) \quad \text{and} \quad v^h(\mathbf{x}, t) = \sum_{j=1}^{N_h} V_j(t) \phi_j(\mathbf{x}), \quad (3.2.34)$$

where  $U_j(t) = u^h(p_j, t)$  and  $V_j(t) = v^h(p_j, t)$ . Without loss of generality, we replace the test functions  $\psi^h(\mathbf{x}, t)$  by  $\phi_i(\mathbf{x}) \in V_h$ ,  $i = 1, 2, \dots, N_h$  and have the following equations:

$$\begin{aligned} & \sum_{j=1}^{N_h} \int \int_{\Omega_h} \phi_i(\mathbf{x}) \cdot \phi_j(\mathbf{x}) \frac{dU_j(t)}{dt} dx dy + \sum_{j=1}^{N_h} \int \int_{\Omega_h} \nabla \phi_i(\mathbf{x}) \cdot \nabla \phi_j(\mathbf{x}) U_j(t) dx dy = \\ & \gamma a \int \int_{\Omega_h} \phi_i(\mathbf{x}) dx dy - \gamma \sum_{j=1}^{N_h} \int \int_{\Omega_h} \phi_i(\mathbf{x}) \cdot \phi_j(\mathbf{x}) U_j(t) dx dy \\ & + \gamma \int \int_{\Omega_h} \phi_i(\mathbf{x}) \left( \sum_{j=1}^{N_h} \phi_j(\mathbf{x}) U_j(t) \right)^2 \sum_{k=1}^{N_h} \phi_k(\mathbf{x}) V_k(t) dx dy, \end{aligned} \quad (3.2.35)$$

and

$$\begin{aligned} & \sum_{j=1}^{N_h} \int \int_{\Omega_h} \phi_i(\mathbf{x}) \cdot \phi_j(\mathbf{x}) \frac{dV_j(t)}{dt} dx dy + d \sum_{j=1}^{N_h} \int \int_{\Omega_h} \nabla \phi_i(\mathbf{x}) \cdot \nabla \phi_j(\mathbf{x}) V_j(t) dx dy = \\ & \gamma b \int \int_{\Omega_h} \phi_i(\mathbf{x}) dx dy - \gamma \int \int_{\Omega_h} \phi_i(\mathbf{x}) \left( \sum_{j=1}^{N_h} \phi_j(\mathbf{x}) U_j(t) \right)^2 \sum_{k=1}^{N_h} \phi_k(\mathbf{x}) V_k(t) dx dy, \end{aligned} \quad (3.2.36)$$

respectively, for all  $i = 1, 2, \dots, N_h$  and  $\mathbf{x} = (x, y)$ . Integrating over the whole computational domain  $\Omega_h$  gives rise to the following semi-discrete equations

$$\begin{cases} \mathbf{M} \frac{d\mathbf{U}(t)}{dt} + \mathbf{A} \mathbf{U}(t) = \gamma a \mathbf{H} - \gamma \mathbf{M} \mathbf{U}(t) + \gamma \mathbf{C}(\mathbf{U}(t), \mathbf{V}(t)), \\ \mathbf{M} \frac{d\mathbf{V}(t)}{dt} + d \mathbf{A} \mathbf{V}(t) = \gamma b \mathbf{H} - \gamma \mathbf{C}(\mathbf{U}(t), \mathbf{V}(t)), \end{cases} \quad (3.2.37)$$

where  $\mathbf{U}(t) = (U_1(t), U_2(t), \dots, U_{N_h}(t))^T$  and  $\mathbf{V}(t) = (V_1(t), V_2(t), \dots, V_{N_h}(t))^T$  are the solution vectors. The matrices  $\mathbf{M}$  and  $\mathbf{A}$  are the global mass matrix and the global stiffness matrix while the vectors  $\mathbf{H}$  and  $\mathbf{C}(\mathbf{U}(t), \mathbf{V}(t))$  are the global force vector and the non-linear vector corresponding to the term  $u^2 v$  respectively. Their entries are

$$\begin{cases} \mathbf{M} = \{m_{ij}\} : m_{ij} = \int \int_{\Omega_h} \phi_i(\mathbf{x}) \cdot \phi_j(\mathbf{x}) dx dy, \\ \mathbf{A} = \{a_{ij}\} : a_{ij} = \int \int_{\Omega_h} \nabla \phi_i(\mathbf{x}) \cdot \nabla \phi_j(\mathbf{x}) dx dy, \\ \mathbf{H} = \{h_i\} : h_i = \int \int_{\Omega_h} \phi_i(\mathbf{x}) dx dy, \\ \mathbf{C} = \{c_i\} : c_i = \int \int_{\Omega_h} \phi_i(\mathbf{x}) \left( \sum_{j=1}^{N_h} \phi_j(\mathbf{x}) U_j(t) \right)^2 \left( \sum_{k=1}^{N_h} \phi_k(\mathbf{x}) V_k(t) \right) dx dy, \end{cases} \quad (3.2.38)$$

respectively for all  $i, j = 1, 2, \dots, N_h$ . We note that the matrices  $\mathbf{M}$  and  $\mathbf{A}$  are symmetric and positive definite.

### Time discretisation

So far we have computed the spatial approximation and ended up with two semi-discrete ODEs. We are now in a position to approximate the system of ODEs in time using finite difference formula for the time derivatives.

We subdivide the time interval  $[0, T]$  into  $N$  uniform subdivisions each of length  $\tau = \frac{T}{N}$ . We denote the initial time by  $t^0$  and the final time by  $t^N$ . Any intermediate time is denoted by  $t^n$ . For the interval  $[0, T]$ , we have  $t^0 = 0$  and  $t^N = T$ . Any point in this interval is therefore given by  $t^n = n\tau$  and the size of each time interval is  $\tau = t^{n+1} - t^n$ . Next, the derivatives with respect to time are approximated by finite differences. As was discussed in Chapter 1, we propose to use the second order semi-implicit backward differentiation formula (2-SBDF) as outlined in Ruuth (1995); Madzvamuse (2006) to (3.2.37). This scheme treats the diffusion terms implicitly and the reaction terms explicitly at time  $t^n$  and  $t^{n-1}$  as follows

$$\begin{cases} \mathbf{M} \left( \frac{3\mathbf{U}^{n+1} - 4\mathbf{U}^n + \mathbf{U}^{n-1}}{2\tau} \right) + \mathbf{A}\mathbf{U}^{n+1} = 2\mathbf{F}^n - \mathbf{F}^{n-1}, \\ \mathbf{M} \left( \frac{3\mathbf{V}^{n+1} - 4\mathbf{V}^n + \mathbf{V}^{n-1}}{2\tau} \right) + d\mathbf{A}\mathbf{V}^{n+1} = 2\mathbf{G}^n - \mathbf{G}^{n-1}, \end{cases}$$

where  $\tau$  is the time-step and  $\mathbf{F}^n$ ,  $\mathbf{F}^{n-1}$ ,  $\mathbf{G}^n$  and  $\mathbf{G}^{n-1}$  represent the discretised reaction kinetics and are shown on the right hand side of (3.2.37). We therefore have the following

$$\begin{cases} \mathbf{M} \left( \frac{3\mathbf{U}^{n+1} - 4\mathbf{U}^n + \mathbf{U}^{n-1}}{2\tau} \right) + \mathbf{A}\mathbf{U}^{n+1} &= 2(\gamma a \mathbf{H} - \gamma \mathbf{M}\mathbf{U}^n + \gamma \mathbf{C}(\mathbf{U}^n, \mathbf{V}^n)) \\ &\quad - (\gamma a \mathbf{H} - \gamma \mathbf{M}\mathbf{U}^{n-1} + \gamma \mathbf{C}(\mathbf{U}^{n-1}, \mathbf{V}^{n-1})), \\ \mathbf{M} \left( \frac{3\mathbf{V}^{n+1} - 4\mathbf{V}^n + \mathbf{V}^{n-1}}{2\tau} \right) + d\mathbf{A}\mathbf{V}^{n+1} &= 2(\gamma b \mathbf{H} - \gamma \mathbf{C}(\mathbf{U}^n, \mathbf{V}^n)) \\ &\quad - (\gamma b \mathbf{H} - \gamma \mathbf{C}(\mathbf{U}^{n-1}, \mathbf{V}^{n-1})). \end{cases}$$

Collecting similar terms and rearranging the equations give the following system of linear equations:

$$\begin{aligned} (3\mathbf{M} + 2\tau\mathbf{A})\mathbf{U}^{n+1} &= 4\mathbf{M}\mathbf{U}^n - \mathbf{M}\mathbf{U}^{n-1} + 2\tau\gamma a \mathbf{H} + 2\tau\gamma \mathbf{M}\mathbf{U}^{n-1} \\ &\quad - 2\tau\gamma \mathbf{C}(\mathbf{U}^{n-1}, \mathbf{V}^{n-1}) - 4\tau\gamma \mathbf{M}\mathbf{U}^n + 4\tau\gamma \mathbf{C}(\mathbf{U}^n, \mathbf{V}^n), \end{aligned} \quad (3.2.39)$$

and

$$\begin{aligned} (3\mathbf{M} + 2\tau d\mathbf{A})\mathbf{V}^{n+1} &= 4\mathbf{M}\mathbf{V}^n - \mathbf{M}\mathbf{V}^{n-1} + 2\tau\gamma b \mathbf{H} \\ &\quad + 2\tau\gamma \mathbf{C}(\mathbf{U}^{n-1}, \mathbf{V}^{n-1}) - 4\tau\gamma \mathbf{C}(\mathbf{U}^n, \mathbf{V}^n). \end{aligned} \quad (3.2.40)$$

Here we notice that we need solutions at both times  $t = t^n$  and  $t = t^{n-1}$ . Solutions for the last two time-steps will therefore need to be stored. To start with, we will use a one step backward Euler method with the reaction terms treated explicitly to solve for the  $\mathbf{U}^1$  and  $\mathbf{V}^1$  solutions as follows

$$\begin{aligned} (\mathbf{M} + \tau\mathbf{A})\mathbf{U}^1 &= \mathbf{M}\mathbf{U}^0 + \tau\gamma a \mathbf{H} - \tau\gamma \mathbf{M}\mathbf{U}^0 + \tau\gamma \mathbf{C}(\mathbf{U}^0, \mathbf{V}^0), \\ (\mathbf{M} + \tau d\mathbf{A})\mathbf{V}^1 &= \mathbf{M}\mathbf{V}^0 + \tau\gamma b \mathbf{H} - \tau\gamma \mathbf{C}(\mathbf{U}^0, \mathbf{V}^0). \end{aligned}$$

and then proceed with the 2-SBDF for all the other time-steps.

### Gaussian quadrature formula

So far we have discretised the space and time and obtained fully discrete system of equations for the Schakenberg model (3.2.8). Before we can solve the system of equations, we will further need to obtain the discrete form of the integrals in the matrices and vectors that we have already set up, namely the mass and stiffness matrices, the global vector and vector representing the non-linear term of the reaction kinetics. To implement the discretisation, we use **deal.II** library [Bangerth et al. \(2007\)](#). Let us begin by reviewing Gaussian quadrature formula very briefly. Consider the canonical quadrilateral element  $\hat{K} = [-1, 1]^2$  in Figure 3.1.

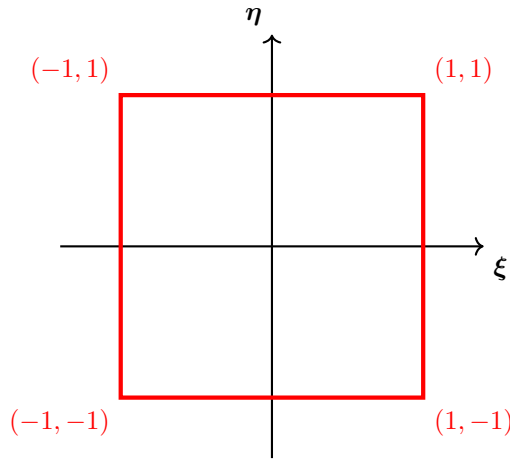
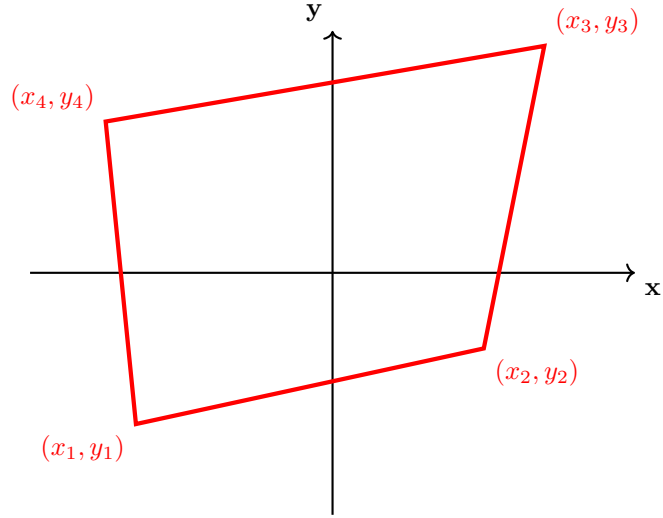


Figure 3.1: The canonical quadrilateral element  $\hat{K}$ .

We write the Gaussian quadrature formula for the function  $g(\xi, \eta)$  on the canonical quadrilateral element  $\hat{K} = [-1, 1]^2$  shown in Figure 3.1 as

$$\begin{aligned} \int_{\hat{K}} g(\xi, \eta) d\xi d\eta &= \int_{-1}^1 \int_{-1}^1 g(\xi, \eta) d\xi d\eta \\ &\approx \sum_{i=1}^N \sum_{j=1}^N w_i \omega_j g(\xi_i, \eta_j), \end{aligned} \tag{3.2.41}$$

where  $w_i, \omega_j$  are the quadrature weights and  $\xi_i, \eta_j$  are the quadrature points in  $\xi$  and  $\eta$  directions respectively. Here, we have used  $N$  quadrature points in each direction. For a general quadrilateral element, we first need to transform it to the canonical element and then apply the quadrature formula (3.2.41). We consider the general quadrilateral element  $K$  in Figure 3.2 with straight boundary lines and vertices  $(x_i, y_i)$ ,  $i = 1, 2, 3, 4$  arranged in anticlockwise manner.

Figure 3.2: A general quadrilateral element  $K$ .

To evaluate the integral

$$I = \iint_K F(x, y) \, dx dy,$$

we first define the nodal shape functions on the canonical quadrilateral element as follows:

$$\begin{cases} \hat{\phi}_1(\xi, \eta) = \frac{1}{4}(1 - \xi)(1 - \eta), \\ \hat{\phi}_2(\xi, \eta) = \frac{1}{4}(1 + \xi)(1 - \eta), \\ \hat{\phi}_3(\xi, \eta) = \frac{1}{4}(1 + \xi)(1 + \eta), \\ \hat{\phi}_4(\xi, \eta) = \frac{1}{4}(1 - \xi)(1 + \eta). \end{cases}$$

We note that  $\hat{\phi}_1$  takes value 1 at the vertex  $(-1, -1)$  and 0 in all the other vertices,  $\hat{\phi}_2$  takes value 1 at the vertex  $(1, -1)$  and 0 in all the other vertices,  $\hat{\phi}_3$  takes value 1 at the vertex  $(1, 1)$  and 0 in all the other vertices and  $\hat{\phi}_4$  takes value 1 at the vertex  $(-1, 1)$  and 0 in all the other vertices and

$$\sum_{i=1}^4 \hat{\phi}_i(\xi, \eta) = 1.$$

We then construct a linear mapping to map the general quadrilateral to the canonical element  $R = [-1, 1]^2$  as shown in Figure 3.3.

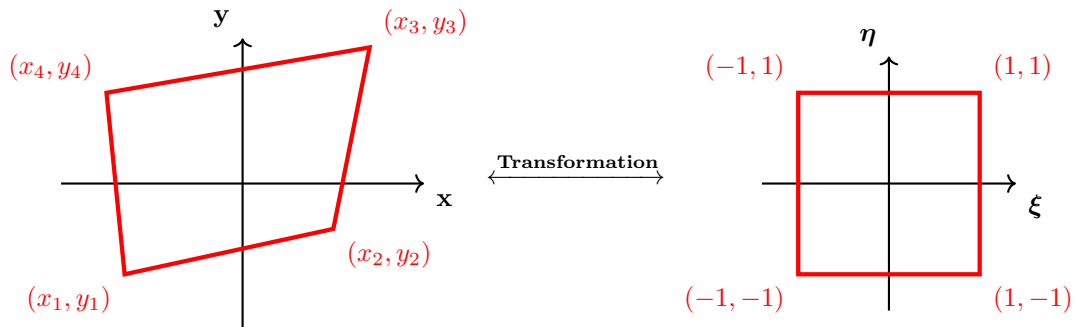


Figure 3.3: Transformation of a general quadrilateral into the canonical quadrilateral element.

The mappings can be achieved conveniently by using the nodal shape functions as follows

$$\begin{cases} x = P(\xi, \eta) = \sum_{i=1}^4 x_i \hat{\phi}_i(\xi, \eta), \\ y = Q(\xi, \eta) = \sum_{i=1}^4 y_i \hat{\phi}_i(\xi, \eta). \end{cases}$$

With this transformation, we write the integral  $I$  as

$$I = \int_K F(x, y) dx dy = \int_{\hat{K}} F(P(\xi, \eta), Q(\xi, \eta)) |\mathbf{J}(\xi, \eta)| d\xi d\eta,$$

where  $|\mathbf{J}(\xi, \eta)|$  is the determinant of the Jacobian of transformation written as

$$|\mathbf{J}(\xi, \eta)| = \begin{vmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} \end{vmatrix}.$$

Applying the quadrature formula (3.2.41) yields

$$\begin{aligned} I &= \int_K F(x, y) dx dy = \int_{\hat{K}} F(P(\xi, \eta), Q(\xi, \eta)) |\mathbf{J}(\xi, \eta)| d\xi d\eta \\ &\approx \sum_{i=1}^N \sum_{j=1}^N w_i \omega_j F(P(\xi_i, \eta_j), Q(\xi_i, \eta_j)) |\mathbf{J}(\xi_i, \eta_j)|, \end{aligned}$$

which is the Gaussian quadrature formula for a general quadrilateral element. Having reviewed the quadrature formula, we now write down the discrete versions of the integrals (3.2.38). We split the integrals over the entire domain into integrals over each cell  $K_i$  and over each cell, discretisation of the integrals are done. This is achieved by mapping back to the reference cell  $\hat{K}$  where all the integrations are carried out. The global matrices and vectors will therefore be given as the contribution over each cell as shown in the equations below

$$\begin{aligned} m_{ij} &= \sum_{K \in \Omega_h} \int_K \phi_i(\mathbf{x}) \cdot \phi_j(\mathbf{x}) dK \\ &= \sum_{K \in \Omega_h} \int_{\hat{K}} \hat{\phi}_i(\hat{\mathbf{x}}) \cdot \hat{\phi}_j(\hat{\mathbf{x}}) |\mathbf{J}_K(\hat{\mathbf{x}})| d\hat{K} \\ &= \sum_{K \in \Omega_h} \sum_q \hat{\phi}_i(\hat{\mathbf{x}}_q) \cdot \hat{\phi}_j(\hat{\mathbf{x}}_q) |\mathbf{J}_K(\hat{\mathbf{x}}_q)| \mathbf{w}(\hat{\mathbf{x}}_q), \end{aligned} \tag{3.2.42}$$

where  $(\hat{\mathbf{x}}_q)$  is the  $q$ th quadrature point on the reference cell  $\hat{K}$  and  $\mathbf{w}(\hat{\mathbf{x}}_q)$  is the corresponding weight. Similarly, for the stiffness matrix  $\mathbf{A}$ , we have

$$\begin{aligned} a_{ij} &= \sum_{K \in \Omega_h} \int_K \nabla \phi_i(\mathbf{x}) \cdot \nabla \phi_j(\mathbf{x}) dK \\ &= \sum_{K \in \Omega_h} \int_{\hat{K}} \hat{\nabla} \hat{\phi}_i(\hat{\mathbf{x}}) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}_q) \cdot \hat{\nabla} \hat{\phi}_j(\hat{\mathbf{x}}) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}_q) |\mathbf{J}_K(\hat{\mathbf{x}})| d\hat{K} \\ &= \sum_{K \in \Omega_h} \sum_q \hat{\nabla} \hat{\phi}_i(\hat{\mathbf{x}}_q) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}_q) \cdot \hat{\nabla} \hat{\phi}_j(\hat{\mathbf{x}}_q) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}_q) |\mathbf{J}_K(\hat{\mathbf{x}}_q)| \mathbf{w}(\hat{\mathbf{x}}_q), \end{aligned} \tag{3.2.43}$$

for the global vector  $\mathbf{H}$ , we have

$$\begin{aligned} h_i &= \sum_{K \in \Omega_h} \int_K \phi_i(\mathbf{x}) dK = \sum_{K \in \Omega_h} \int_{\hat{K}} \hat{\phi}_i(\hat{\mathbf{x}}) |\mathbf{J}_K(\hat{\mathbf{x}})| d\hat{K} \\ &= \sum_{K \in \Omega_h} \sum_q \hat{\phi}_i(\hat{\mathbf{x}}_q) |\mathbf{J}_K(\hat{\mathbf{x}}_q)| \mathbf{w}(\hat{\mathbf{x}}_q), \end{aligned} \tag{3.2.44}$$



and for the vector  $\mathbf{C}$ , we have

$$\begin{aligned}
c_i &= \sum_{K \in \Omega_h} \int_K \phi_i(\mathbf{x}) \left( \sum_{j=1}^{N_h} \phi_j(\mathbf{x}) U_j(t) \right)^2 \left( \sum_{k=1}^{N_h} \phi_k(\mathbf{x}) V_k(t) \right) dK \\
&= \sum_{K \in \Omega_h} \int_{\hat{K}} \hat{\phi}_i(\hat{\mathbf{x}}) \left( \sum_{j=1}^{N_h} \hat{\phi}_j(\hat{\mathbf{x}}) U_j(t) \right)^2 \left( \sum_{k=1}^{N_h} \hat{\phi}_k(\hat{\mathbf{x}}) V_k(t) \right) |\mathbf{J}_K(\hat{\mathbf{x}})| d\hat{K} \\
&= \sum_{K \in \Omega_h} \sum_q \hat{\phi}_i(\hat{\mathbf{x}}_q) \left( \sum_{j=1}^{N_h} \hat{\phi}_j(\hat{\mathbf{x}}_q) U_j(t) \right)^2 \left( \sum_{k=1}^{N_h} \hat{\phi}_k(\hat{\mathbf{x}}_q) V_k(t) \right) |\mathbf{J}_K(\hat{\mathbf{x}}_q)| \mathbf{w}(\hat{\mathbf{x}}_q).
\end{aligned} \tag{3.2.45}$$

To implement the discretisation, we use **deal.II** library [Bangerth et al. \(2007\)](#) which is an efficient finite element library written in **C++** language. It is an open-source library and provides tools to solve problems that specifically use the finite element method. We note that both the mass and stiffness matrices  $\mathbf{M}$  and  $\mathbf{A}$  respectively are large, sparse, symmetric and positive definite. A sparse matrix has very few nonzero entries. This suggests that the classical direct methods for solving these systems will not be efficient. Iterative methods are therefore the best option [Saad \(2003\)](#). By iterative methods, we refer to a wide range of techniques that use successive approximations to obtain a more accurate solution to a linear system at each time step. Here, we will use iterative methods that use pre-conditioners. The rate of convergence of an iterative method depends greatly on the spectrum of the matrices, that is, the rate of convergence depends on the condition number of the matrices  $\mathbf{M}$  and  $\mathbf{A}$  [Saad \(2003\)](#). A pre-conditioner is a transformation matrix that when applied to the matrices  $\mathbf{M}$  or  $\mathbf{A}$  will give rise to a more favourable spectrum of the resultant matrices [Barrett et al. \(1994\)](#); [Madzvamuse \(2000\)](#). Therefore a good pre-conditioner will improve the rate of convergence of the iterative method [Barrett et al. \(1994\)](#). Since the matrices  $\mathbf{M}$  and  $\mathbf{A}$  are both symmetric, Equations (3.2.39) and (3.2.40) can be solved using a preconditioned conjugate gradient method (PCG) [Saad \(2003\)](#). This method requires that the matrices  $\mathbf{M}$ ,  $\mathbf{A}$  and preconditioners be symmetric and positive definite. For these matrices, we will choose the diagonal matrices  $\text{diag}(3\mathbf{M} + 2\tau\mathbf{A})$  and  $\text{diag}(3\mathbf{M} + 2\tau d\mathbf{A})$  as the pre-conditioners [Saad \(2003\)](#). Since both matrices  $\mathbf{M}$  and  $\mathbf{A}$  are symmetric and positive definite, so are the diagonal matrices.

## Numerical simulations

We now present some numerical simulations for the Schnakenberg model. For our simulations, the domain  $\Omega$  is the unit disk. We discretise the unit disk into 5120 quadrilaterals (elements) with 5185 degrees of freedom. The initial data is chosen to be a random perturbations from the equilibrium points (3.2.9). In all the simulations, we choose parameters as displayed in Table 3.2. To check the stability, consistency and convergence of the numerical solutions of the model, we verify computationally that the errors between successive finite element approximate solutions decay with time and that the numerical solutions are bounded. We do so by computing the  $L_2$ -norm  $\|\frac{u^{n+1}-u^n}{\tau}\|$  for the solutions of the  $u$  variable and similarly for the  $v$  variable  $\|\frac{v^{n+1}-v^n}{\tau}\|$ . For comparison of the numerical results to results from linear stability analysis, we refer to [Murray \(2002\)](#); [Madzvamuse \(2000\)](#); [Chung \(2016\)](#); [George \(2012\)](#). The simulations were allowed to run

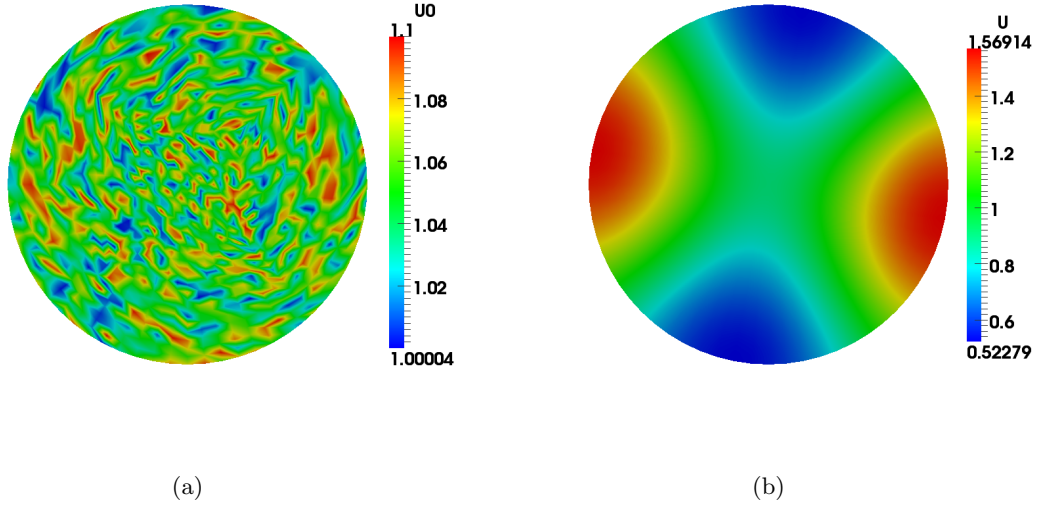


Figure 3.4: Solutions for the  $u$  variable of the Schnakenberg model using a 2-SBDF scheme with  $\tau = 2 \times 10^{-3}$ . Blue signifies lowest values while red highest values. (a) Initial condition as random perturbations about steady states for the  $u$  variable and (b) solution at a final time step  $t = 10$  showing convergence to an in-homogeneous steady state. Parameters values used are  $a = 0.1$ ,  $b = 0.9$ ,  $d = 10$  and  $\gamma = 29$ .

until a spatially in-homogeneous steady state was reached as shown in Figures 3.4, 3.5 and 3.6. At each time step, we carry out multiple iterations and terminate computation subject to the condition that

$$\|u^{k+1} - u^k\|_2 \leq \text{Tol},$$

where  $k$  is number of iterations and (Tol) is some acceptable tolerance between two successive iterations. We chose different time-step sizes and compared the  $L_2$ -norm  $\|\frac{u^{n+1}-u^n}{\tau}\|$  for the solutions for  $u$  variable and similarly for the  $v$  variable  $\|\frac{v^{n+1}-v^n}{\tau}\|$ . Furthermore, for a particular  $\tau$ , we varied the mesh size and compared the  $L_2$ -norm. To check the stability region, we computed the relative error given by

$$\text{Relative error} = \sqrt{\frac{\sum |U^{n+1} - U^n|^2}{\sum |U^{n+1}|^2}}, \quad (3.2.46)$$

at the final time  $t = 10$  and displayed the results in Table 3.1. We also varied the parameter  $\gamma$  and plot the solutions in Figure 3.6. We plot the graph for the  $L_2$ -norm with time as shown in Figures 3.7 and 3.8.

### Discussion of the numerical results

The simulations began with initial conditions as random perturbations about the steady states (3.2.9). Simulation was allowed to run until time  $t = 10$  which was long enough to achieve spatially varying steady states for the variables  $u$  and  $v$ . The  $L_2$ -norm is related to the rate of change of the variables. We note that the  $L_2$ -norms decay with time indicating convergence of the numerical

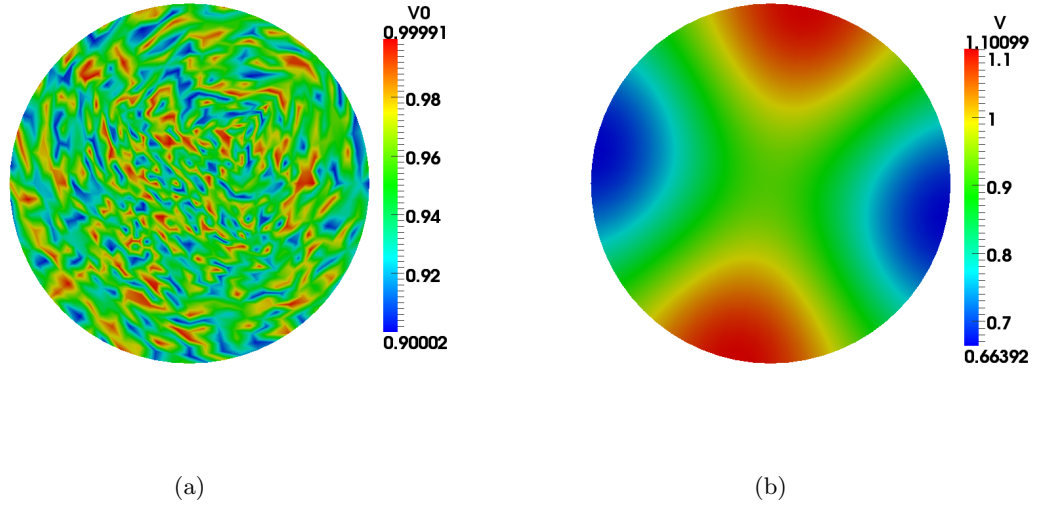


Figure 3.5: Solutions for the  $v$  variable of the Schnakenberg model using a 2-SBDF scheme with  $\tau = 2 \times 10^{-3}$ . Blue signifies lowest values while red highest values. (a) Initial condition as random perturbations about steady states for the  $v$  variable and (b) solution at a final time step  $t = 10$  showing convergence to an in-homogeneous steady state. Parameters values used are  $a = 0.1$ ,  $b = 0.9$ ,  $d = 10$  and  $\gamma = 29$ .

Time-step $\tau$	No. of time steps	Relative error	$\frac{\ \mathbf{u}^{n+1} - \mathbf{u}^n\ }{\tau}$
$2.0 \times 10^{-2}$	500	0.157128	518.582
$1.85 \times 10^{-2}$	540	0.00239478	3.81838
$1.82 \times 10^{-2}$	550	$8.79775 \times 10^{-7}$	0.00522085
$1.33 \times 10^{-2}$	750	$2.58141 \times 10^{-7}$	0.00141715
$10^{-2}$	1,000	$2.34805 \times 10^{-7}$	0.00171531
$2.0 \times 10^{-3}$	5,000	$6.51356 \times 10^{-8}$	0.00237928
$10^{-3}$	10,000	$3.30758 \times 10^{-8}$	0.0024164

Table 3.1: Convergence of the  $u$  variable using the 2-SBDF scheme at different time steps  $\tau$  showing the effects of time-step refinement on the magnitudes of errors.

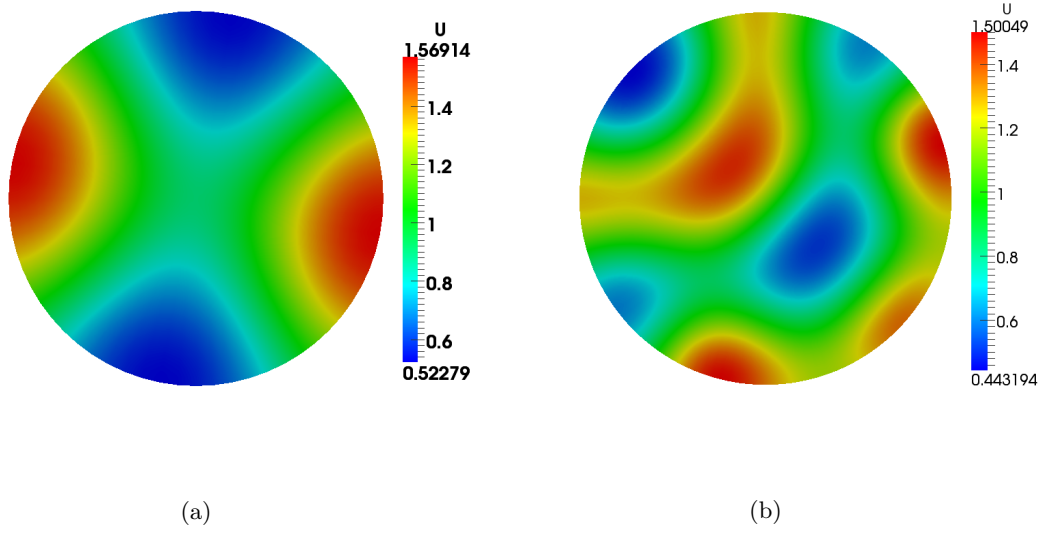


Figure 3.6: Solutions for the  $u$  variable of the Schnakenberg model with  $\tau = 2 \times 10^{-3}$ ,  $a = 0.1$ ,  $b = 0.9$ ,  $d = 10$  and (a)  $\gamma = 29$  and (b)  $\gamma = 100$  showing convergence to in-homogeneous steady states. Blue signifies lowest values while red highest values.

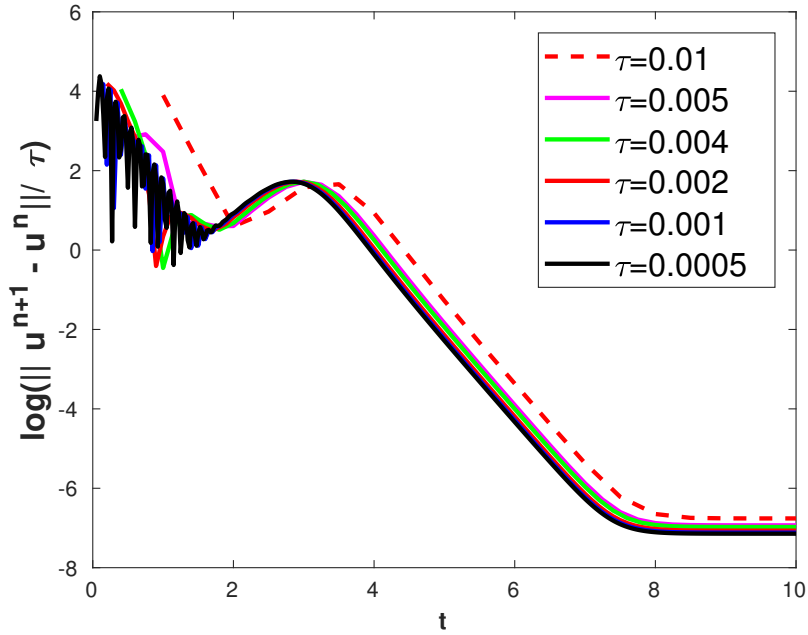


Figure 3.7: Convergence history of the simulations of the Schnakenberg model using a 2-SBDF scheme for the  $u$  variable with time refinements.

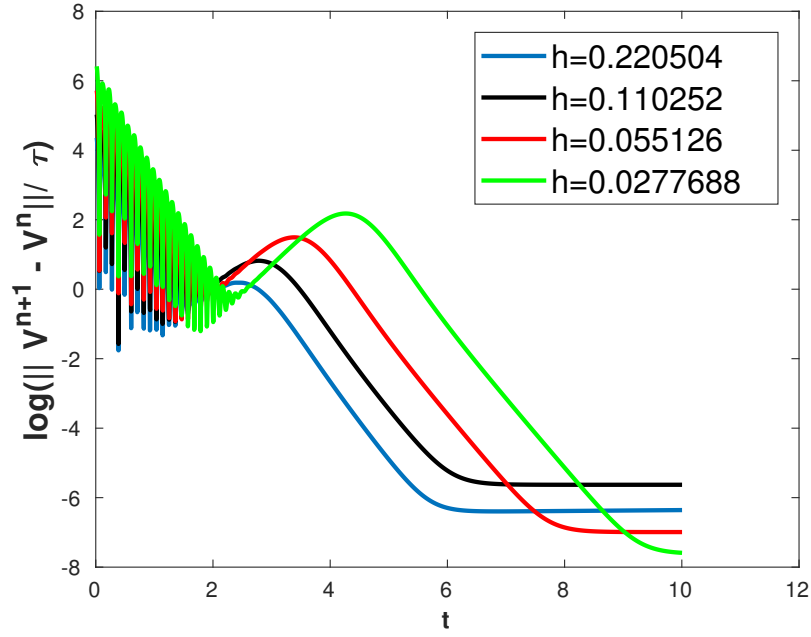


Figure 3.8: Convergence history of the simulations of the Schnakenberg model for the  $v$  variable with mesh refinements.

$a$	$b$	$d$	$\gamma$
0.1	0.9	10	29, 100

Table 3.2: The parameters for the Schnakenberg model.

solutions. We observed three key phases in the numerical process as shown in the plots for the  $L_2$ -norm. These are initial decay, followed by growth and then again decay as the solution converge to the spatially varying steady states. There was a significant effect of time-step refinement on the numerical solutions and the magnitudes of errors as seen in Table 3.1 and Figure 3.7. There was no significant effect of mesh refinement on the numerical solutions.

Our results validate the theoretical results of diffusion-driven instability (Turing instability) which states that in the absence of diffusion, equilibrium point (3.2.9) is linearly stable provided that condition (3.2.12) is satisfied but if one then allows spatial inhomogeneity, it is possible that the system evolves to an in-homogeneous steady state. We began with initial condition as random perturbation about steady states for both the  $u$  and  $v$  variables and solutions evolved to in-homogeneous steady states. For diffusion-driven instability to be achieved, parameters need to be selected from a space called Turing space Murray (2002).

## Summary

We have considered the well known Schnakenberg model. We derived the conditions for diffusion-driven instability as indicated in Murray (2002); Turing (1952); Madzvamuse (2000). Since the model is highly nonlinear, we went ahead and sought numerical solution using finite elements.

We dealt with the weak formulation of the model, then described the standard finite element method. In our case, we used the Galerkin method which uses the same space of functions for both the approximate finite element solutions and the test functions. The construction of the finite element method was done using quadrilateral elements to discretise the space. To discretise the time, we used the 2-SBDF method. The resulting discrete system of equations was solved using preconditioned conjugate gradient method. The solution for the discrete system is the approximate solution of the  $u$  and  $v$  variables.

In summary, our results validate the theoretical results of diffusion-driven instability which states that in the absence of diffusion, equilibrium point (3.2.9) is linearly stable provided that condition (3.2.12) is satisfied but if one then allows spatial inhomogeneity, it is possible that the system evolves to an in-homogeneous steady state. We began with initial conditions as random perturbation about steady states for both the  $u$  and  $v$  variables and solutions evolved to in-homogeneous steady states. Now that we have demonstrated that our solver works for the standard and well known Schnakenberg model, we apply it to the viscous model for cell migration.

### 3.2.3 The viscous model for cell migration

In this section, we aim to solve the full model for cell migration that was formulated in the previous chapter. We will use the unit disk as our domain representing the cell at the initial time. We will apply the finite element method that we discussed to discretise the space and the second order semi-implicit backward differentiation formula (2-SBDF), see Ruuth (1995); Madzvamuse (2006) for the time discretisation to transform the system of partial differential equations to a system of algebraic equations. We will first consider the model equations on a stationary cell and solve the biochemical model for F-actin and myosin II. Once we have solved the biochemical model, we will incorporate the force balance equation in order to solve the full model for cell migration. We begin by rewriting the non-dimensionalised viscous model for cell migration.

#### The non-dimensionalised viscous model for cell migration

$$\left\{ \begin{array}{l} \frac{\partial \rho_m}{\partial t} + b \nabla \cdot (\rho_m \beta) = \nabla \cdot \left( \left( \frac{1}{1+a\rho_a} \right) \nabla \rho_m \right), \quad \mathbf{x} \in \Omega_t, \quad t \in (0, T], \\ \frac{\partial \rho_a}{\partial t} + b \nabla \cdot (\rho_a \beta) = d \Delta \rho_a + \left( \frac{k_3 \rho_a^2}{1+\rho_a^2} + k_4 \right) \rho_a^{cyl} - e \rho_a, \quad \mathbf{x} \in \Omega_t, \quad t \in (0, T], \\ \nabla \cdot (\sigma_\nu(\mathbf{x}, t) + \sigma_{myo}(\mathbf{x}, t) + \sigma_{poly}(\mathbf{x}, t)) = \mathbf{0}, \quad \mathbf{x} \in \Omega_t, \quad t \in (0, T], \\ \rho_a(\mathbf{x}, 0) = \rho_a^0(\mathbf{x}), \quad \rho_m(\mathbf{x}, 0) = \rho_m^0(\mathbf{x}), \quad \mathbf{x} \in \Omega_t, \quad t = 0, \\ \sigma_\nu \cdot \mathbf{n} = \mathbf{0}, \quad \mathbf{x} \in \partial \Omega_t, \quad t \in (0, T], \\ \frac{\partial \rho_a}{\partial \mathbf{n}} = \frac{\partial \rho_m}{\partial \mathbf{n}} = 0, \quad \mathbf{x} \in \partial \Omega_t, \quad t \in (0, T], \end{array} \right. \quad (3.2.47)$$

with

$$\begin{cases} \rho_a^{cyl}(t) = \frac{\rho_a^{tot} - \int_{\Omega_t} \rho_a d\Omega_t}{\int_{\Omega_t} d\Omega_t}, \\ \sigma_{myo}(\mathbf{x}, t) = \eta_1 \rho_m(\mathbf{x}, t) \mathbf{I}, \quad \eta_1 \in \mathbb{R}^+, \\ \sigma_\nu(\mathbf{x}, t) = \nabla \beta(\mathbf{x}, t) + (\nabla \beta(\mathbf{x}, t))^T, \\ \sigma_{poly}(\mathbf{x}, t) = -\eta_2 \rho_a(\mathbf{x}, t) \delta(l) \mathbf{I}, \quad \eta_2 \in \mathbb{R}^+. \end{cases} \quad (3.2.48)$$

### Biochemical model for F-actin and myosin II

We consider a stationary domain  $\Omega \subset \mathbb{R}^2$  representing the cell. This means that we consider the case  $\beta(\mathbf{x}, t) = \mathbf{0}$  in (3.2.47). The reaction-diffusion equations for actin-myosin system on  $\Omega$  now reads

$$\begin{cases} \frac{\partial \rho_m}{\partial t} = \nabla \cdot \left( \left( \frac{1}{1+a\rho_a} \right) \nabla \rho_m \right), \quad \mathbf{x} \in \Omega, \quad t \in (0, T], \\ \rho_m(\mathbf{x}, 0) = \rho_m^0(\mathbf{x}), \quad \mathbf{x} \in \Omega, \quad t = 0, \\ \frac{\partial \rho_m}{\partial \mathbf{n}} = 0, \quad \mathbf{x} \in \partial\Omega, \quad t \in (0, T], \end{cases} \quad (3.2.49)$$

and

$$\begin{cases} \frac{\partial \rho_a}{\partial t} = d\Delta \rho_a + \left( \frac{k_3 \rho_a^2}{1+\rho_a^2} + k_4 \right) \rho_a^{cyl} - e\rho_a, \quad \mathbf{x} \in \Omega, \quad t \in (0, T], \\ \rho_a(\mathbf{x}, 0) = \rho_a^0(\mathbf{x}), \quad \mathbf{x} \in \Omega, \quad t = 0, \\ \frac{\partial \rho_a}{\partial \mathbf{n}} = 0, \quad \mathbf{x} \in \partial\Omega, \quad t \in (0, T], \end{cases} \quad (3.2.50)$$

with

$$\rho_a^{cyl} = \frac{\rho_a^{tot} - \int_{\Omega} \rho_a d\Omega}{\int_{\Omega} d\Omega},$$

which is the biochemical model for myosin II and F-actin respectively.

### Weak formulation for the reaction-diffusion equation on a stationary domain

To derive the weak formulation of the reaction-diffusion equation (3.2.49) and (3.2.50), we multiply by a test function say  $\psi_1(\mathbf{x}, t)$   $\psi_2(\mathbf{x}, t) \in H^1(\Omega)$  respectively and integrate over  $\Omega$  as follows

$$\int_{\Omega} \psi_1 \frac{\partial \rho_m}{\partial t} d\Omega = \int_{\Omega} \psi_1 \nabla \cdot (D_m(\rho_a) \nabla \rho_m) d\Omega, \quad (3.2.51)$$

$$\int_{\Omega} \psi_2 \frac{\partial \rho_a}{\partial t} d\Omega = \int_{\Omega} (d\psi_2 \Delta \rho_a + \psi_2 f(\rho_a, \rho_a^{cyl})) d\Omega. \quad (3.2.52)$$

Now we apply Green's formula to the above equations and use  $\frac{\partial \rho_m}{\partial \mathbf{n}} = 0$  and  $\frac{\partial \rho_a}{\partial \mathbf{n}} = 0$  on  $\partial\Omega$ .

This gives the following weak formulation: find  $\rho_m(\mathbf{x}, t)$ ,  $\rho_a(\mathbf{x}, t) \in H^1(\Omega)$ ,  $t \in (0, T]$  such that

$$\int_{\Omega} \psi_1 \frac{\partial \rho_m}{\partial t} d\Omega = - \int_{\Omega} D_m(\rho_a) \nabla \psi_1 \cdot \nabla \rho_m d\Omega, \quad (3.2.53)$$

$$\int_{\Omega} \psi_2 \frac{\partial \rho_a}{\partial t} d\Omega = \int_{\Omega} (-d\nabla \psi_2 \cdot \nabla \rho_a + \psi_2 f(\rho_a, \rho_a^{cyl})) d\Omega, \quad (3.2.54)$$

for all  $\psi_1(\mathbf{x}, t)$ ,  $\psi_2(\mathbf{x}, t) \in H^1(\Omega)$ , where

$$\begin{aligned} D_m(\rho_a) &= \frac{1}{1+a\rho_a}, \\ f(\rho_a, \rho_a^{cyl}) &= \left( \frac{k_3 \rho_a^2}{1+\rho_a^2} + k_4 \right) \rho_a^{cyl} - e\rho_a, \\ \rho_a^{cyl} &= \frac{\rho_a^{tot} - \int_{\Omega} \rho_a d\Omega}{\int_{\Omega} d\Omega}. \end{aligned}$$

### Space discretisation

To be able to apply a numerical method, the continuous domain needs to be discretised and only be defined at some finite number of points. We do this by using finite elements which is well adapted even for irregularly-shaped domains. Let  $\Omega_h$  be the computational domain which is a polyhedral approximation to  $\Omega$ . We define  $T_h$  to be a triangulation of  $\Omega_h$  made up of non-degenerate rectangular elements  $K_i$  such that  $T_h = \bigcup_i K_i$ . We call each  $K_i$  an element of the mesh  $T_h$  where  $h$  is the diameter of the largest element. For the mesh  $T_h$ , we require that it is made up of a finite number of elements and the elements must intersect along a complete edge, or at a vertex or not at all. The space discretisation is carried out using quadrilateral elements and a piece-wise linear approximation of the solution is sought. We define the finite element space  $V_h \subset \mathbf{H}^1(\Omega)$  by

$$V_h = \{v_h \in C^0(\Omega) : v_h|_K \text{ is linear}\} \quad (3.2.55)$$

We will seek solutions of the reaction-diffusion equations (3.2.49) and (3.2.50) in this space. The discretised version of the weak forms of the reaction diffusion equations above therefore reads: find  $\rho_m^h(\mathbf{x}, t), \rho_a^h(\mathbf{x}, t) \in V_h$  such that

$$\int_{\Omega_h} \psi_1^h \frac{\partial \rho_m^h}{\partial t} d\Omega_h = - \int_{\Omega_h} D_m(\rho_a^h) \nabla \psi_1^h \cdot \nabla \rho_m^h d\Omega_h, \quad (3.2.56)$$

$$\int_{\Omega_h} \psi_2^h \frac{\partial \rho_a^h}{\partial t} d\Omega_h = \int_{\Omega_h} (-\nabla \psi_2^h \cdot \nabla \rho_a^h + \psi_2^h f(\rho_a^h)) d\Omega_h, \quad (3.2.57)$$

for all  $\psi_1^h(\mathbf{x}, t), \psi_2^h(\mathbf{x}, t) \in V_h$ , where

$$D_m(\rho_a^h) = \frac{1}{1 + a\rho_a^h},$$

$$f(\rho_a^h, \rho_a^{cyl,h}) = \left( \frac{k_3(\rho_a^h)^2}{1 + (\rho_a^h)^2} + k_4 \right) \rho_a^{cyl,h} - e\rho_a^h,$$

$$\rho_a^{cyl,h} = \frac{\rho_a^{tot} - \int_{\Omega_h} \rho_a^h d\Omega_h}{\int_{\Omega_h} d\Omega_h}.$$

Since  $V_h$  is a linear space, it must have a basis. We introduce the basis function

$$\phi_i(\mathbf{x}) \in V_h, \quad i = 1, 2, \dots, N_h, \quad (3.2.58)$$

such that

$$\phi_i(p_j) = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{if } i \neq j, \end{cases} \quad (3.2.59)$$

where  $p_j$  is the  $j$ th nodal point of the mesh. We seek to find the finite element numerical approximations of the form  $\rho_m^h(\mathbf{x}, t), \rho_a^h(\mathbf{x}, t) \in V_h$  expressed as linear combinations of the linear nodal basis functions  $\phi_i(\mathbf{x})$  as follows

$$\rho_m^h(\mathbf{x}, t) = \sum_{j=1}^{N_h} \omega_j(t) \phi_j(\mathbf{x}) \quad \text{and} \quad \rho_a^h(\mathbf{x}, t) = \sum_{j=1}^{N_h} \rho_j(t) \phi_j(\mathbf{x}), \quad (3.2.60)$$



where  $\omega_j(t) = \rho_m^h(p_j, t)$  and  $\rho_j(t) = \rho_a^h(p_j, t)$ . Without loss of generality, we replace the test functions  $\psi_1^h(\mathbf{x}, t)$ ,  $\psi_2^h(\mathbf{x}, t)$  by  $\phi_i(\mathbf{x}) \in V_h$ ,  $i = 1, 2, \dots, N_h$  and have the equations approximated by

$$\sum_{j=1}^{N_h} \int_{\Omega_h} \phi_i(\mathbf{x}) \cdot \phi_j(\mathbf{x}) \frac{d\omega_j(t)}{dt} d\Omega_h = - \sum_{j=1}^{N_h} \int_{\Omega_h} \frac{\nabla \phi_i(\mathbf{x}) \cdot \nabla \phi_j(\mathbf{x}) \omega_j(t)}{1 + a \sum_{k=1}^{N_h} \rho_k(t) \phi_k(\mathbf{x})} d\Omega_h, \quad (3.2.61)$$

and

$$\begin{aligned} \sum_{j=1}^{N_h} \int_{\Omega_h} \phi_i(\mathbf{x}) \cdot \phi_j(\mathbf{x}) \frac{d\phi_j(t)}{dt} d\Omega_h &= -d \sum_{j=1}^{N_h} \int_{\Omega_h} \nabla \phi_i(\mathbf{x}) \cdot \nabla \phi_j(\mathbf{x}) \rho_j(t) d\Omega_h \\ &\quad - e \sum_{j=1}^{N_h} \int_{\Omega_h} \phi_i(\mathbf{x}) \cdot \phi_j(\mathbf{x}) \rho_j(t) d\Omega_h \\ &\quad + k_4 a_{mon}(t) \int_{\Omega_h} \phi_i(\mathbf{x}) d\Omega_h \\ &\quad + k_3 a_{mon}(t) \int_{\Omega_h} \frac{\phi_i(\mathbf{x}) \left( \sum_{j=1}^{N_h} \rho_j(t) \phi_j(\mathbf{x}) \right)^2}{1 + \left( \sum_{j=1}^{N_h} \rho_j(t) \phi_j(\mathbf{x}) \right)^2} d\Omega_h, \end{aligned} \quad (3.2.62)$$

respectively, for all  $i = 1, 2, \dots, N_h$  where  $\mathbf{x} = (x, y)$ . The parameter  $a_{mon}(t)$  represents the well mixed actin monomers concentration. Integrating over the whole computational domain  $\Omega_h$  gives rise to the following semi-discrete equations

$$\mathbf{M} \frac{d\boldsymbol{\omega}(t)}{dt} = -\mathbf{P}(t) \boldsymbol{\omega}(t), \quad (3.2.63)$$

and

$$\mathbf{M} \frac{d\boldsymbol{\rho}(t)}{dt} = -d\mathbf{A}\boldsymbol{\rho}(t) - e\mathbf{M}\boldsymbol{\rho}(t) + k_4 a_{mon}(t) \mathbf{H} + k_3 a_{mon}(t) \mathbf{L}(t), \quad (3.2.64)$$

where  $\boldsymbol{\omega}(t) = (\omega_1(t), \omega_2(t), \dots, \omega_{N_h}(t))^T$  and  $\boldsymbol{\rho}(t) = (\rho_1(t), \rho_2(t), \dots, \rho_{N_h}(t))^T$  are the solution vectors. Here,  $\mathbf{M}$  is the global mass matrix,  $\mathbf{A}$  is the global stiffness matrix,  $\mathbf{H}$  is the global force vector,  $\mathbf{P}(t)$  is a variable matrix while  $\mathbf{L}(t)$  is a variable vector. The mass and stiffness matrices are respectively given by

$$\mathbf{M} = \{m_{ij}\} : m_{ij} = \int_{\Omega_h} \phi_i(\mathbf{x}) \cdot \phi_j(\mathbf{x}) d\Omega_h,$$

$$\mathbf{A} = \{a_{ij}\} : a_{ij} = \int_{\Omega_h} \nabla \phi_i(\mathbf{x}) \cdot \nabla \phi_j(\mathbf{x}) d\Omega_h,$$

which are symmetric and positive definite. The global force vector is given by

$$\mathbf{H} = \{h_i\} : h_i := \int_{\Omega_h} \phi_i(\mathbf{x}) d\Omega_h,$$

while the  $\rho_a$  dependent matrix is given by

$$\mathbf{P} = \{p_{ij}\} : p_{ij} = \int_{\Omega_h} \frac{\nabla \phi_i(\mathbf{x}) \cdot \nabla \phi_j(\mathbf{x})}{1 + a \sum_{k=1}^{N_h} \rho_k(t) \phi_k(\mathbf{x})} d\Omega_h,$$

and is non-symmetric. The nonlinear vector is given by

$$\mathbf{L} = \{l_i\} : l_i = \int_{\Omega_h} \frac{\phi_i(\mathbf{x}) \left( \sum_{k=1}^{N_h} \rho_k(t) \phi_k(\mathbf{x}) \right)^2}{1 + \left( \sum_{k=1}^{N_h} \rho_k(t) \phi_k(\mathbf{x}) \right)^2} d\Omega_h.$$

### Time discretisation

So far, we have computed the spatial approximation and ended up with two semi-discrete ODEs. We can now approximate the system of ODEs in time using finite differences for the time derivatives. We subdivide the time interval  $(0, T]$  into  $N$  uniform subdivisions each of length  $\tau = \frac{T}{N}$ . We denote the initial time by  $t^0$  and the final time by  $t^N$ . Any intermediate time is denoted by  $t^n$ . We employ the second order semi-implicit backward differentiation formula as follows

$$\mathbf{M} \left( \frac{3\boldsymbol{\omega}^{n+1} - 4\boldsymbol{\omega}^n + \boldsymbol{\omega}^{n-1}}{2\tau} \right) = -\mathbf{P}(t^n)\boldsymbol{\omega}^{n+1},$$

and

$$\begin{aligned} \mathbf{M} \left( \frac{3\boldsymbol{\rho}^{n+1} - 4\boldsymbol{\rho}^n + \boldsymbol{\rho}^{n-1}}{2\tau} \right) = & -d\mathbf{A}\boldsymbol{\rho}^{n+1} + 2(-e\mathbf{M}\boldsymbol{\rho}^n + k_4a_{mon}(t^n)\mathbf{H} + k_3a_{mon}(t^n)\mathbf{L}(t^n)) \\ & - (-e\mathbf{M}\boldsymbol{\rho}^{n-1} + k_4a_{mon}(t^{n-1})\mathbf{H} + k_3a_{mon}(t^{n-1})\mathbf{L}(t^{n-1})). \end{aligned}$$

Simplifying the above system of equations yields

$$(3\mathbf{M} + 2\tau\mathbf{P}(t^n))\boldsymbol{\omega}(t^{n+1}) = 4\mathbf{M}\boldsymbol{\omega}(t^n) - \mathbf{M}\boldsymbol{\omega}(t^{n-1}), \quad (3.2.65)$$

and

$$\begin{aligned} (3\mathbf{M} + 2d\tau\mathbf{A})\boldsymbol{\rho}(t^{n+1}) = & 4\mathbf{M}\boldsymbol{\rho}(t^n) - \mathbf{M}\boldsymbol{\rho}^{n-1} \\ & + 4\tau(-e\mathbf{M}\boldsymbol{\rho}^n + k_4a_{mon}(t^n)\mathbf{H} + k_3a_{mon}(t^n)\mathbf{L}(t^n)) \\ & - 2\tau(-e\mathbf{M}\boldsymbol{\rho}^{n-1} + k_4a_{mon}(t^{n-1})\mathbf{H} + k_3a_{mon}(t^{n-1})\mathbf{L}(t^{n-1})). \end{aligned} \quad (3.2.66)$$

where  $\boldsymbol{\omega}^n, \boldsymbol{\omega}^{n+1}$  are the myosin II solutions at time  $t^n$  and  $t^{n+1}$  respectively while  $\boldsymbol{\rho}^n, \boldsymbol{\rho}^{n+1}$  are the F-actin solutions at time  $t^n$  and  $t^{n+1}$  respectively. Here, we need solutions at both times  $t = t^n$  and  $t = t^{n-1}$ . Solutions for the last two time-steps will therefore need to be stored. To start with, we will use a one step backward Euler method where the reaction terms are treated explicitly to solve for the  $\boldsymbol{\omega}(t^1)$  and  $\boldsymbol{\rho}(t^1)$  solutions as follows

$$\begin{aligned} (\mathbf{M} + \tau\mathbf{P}(t^0))\boldsymbol{\omega}(t^1) &= \mathbf{M}\boldsymbol{\omega}(t^0), \\ (\mathbf{M} + d\tau\mathbf{A})\boldsymbol{\rho}(t^1) &= \mathbf{M}\boldsymbol{\rho}(t^0) + \tau(-e\mathbf{M}\boldsymbol{\rho}(t^0) + k_4a_{mon}(t^0)\mathbf{H} + k_3a_{mon}(t^0)\mathbf{L}(t^0)), \end{aligned}$$

and then proceed with the 2-SBDF for all the other time-steps.

### Quadrature formula

Before we are able to implement an algorithm to solve the discrete system above, we further need to discretise the integrals in the matrices and vectors. To implement the discretisation, we use **deal.II** library [Bangerth et al. \(2007\)](#). Let each quadrilateral be mapped onto the canonical quadrilateral  $\hat{K}$  by a bi-linear transformation. We split the integrals over the entire domain into integrals over each cell  $K_i$  and map each cell to  $\hat{K}$ . The global matrices and vectors will therefore be given as

the contribution over each cell. The quadrature formula for the mass matrix is given by

$$\begin{aligned}
m_{ij} &= \sum_{K \in \Omega_h} \int_K \phi_i(\mathbf{x}) \cdot \phi_j(\mathbf{x}) dK \\
&= \sum_{K \in \Omega_h} \int_{\hat{K}} \hat{\phi}_i(\hat{\mathbf{x}}) \cdot \hat{\phi}_j(\hat{\mathbf{x}}) |\mathbf{J}_K(\hat{\mathbf{x}})| d\hat{K} \\
&= \sum_{K \in \Omega_h} \sum_q \hat{\phi}_i(\hat{\mathbf{x}}_q) \cdot \hat{\phi}_j(\hat{\mathbf{x}}_q) |\mathbf{J}_K(\hat{\mathbf{x}}_q)| \mathbf{w}(\hat{\mathbf{x}}_q),
\end{aligned} \tag{3.2.67}$$

where  $(\hat{\mathbf{x}}_q)$  is the  $q$ th quadrature point on the reference cell  $\hat{K}$  and  $\mathbf{w}(\hat{\mathbf{x}}_q)$  is the corresponding weight. Similarly, the quadrature formula for the stiffness matrix is

$$\begin{aligned}
a_{ij} &= \sum_{K \in \Omega_h} \int_K \nabla \phi_i(\mathbf{x}) \cdot \nabla \phi_j(\mathbf{x}) dK \\
&= \sum_{K \in \Omega_h} \int_{\hat{K}} \hat{\nabla} \hat{\phi}_i(\hat{\mathbf{x}}) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}) \cdot \hat{\nabla} \hat{\phi}_j(\hat{\mathbf{x}}) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}) |\mathbf{J}_K(\hat{\mathbf{x}})| d\hat{K} \\
&= \sum_{K \in \Omega_h} \sum_q \hat{\nabla} \hat{\phi}_i(\hat{\mathbf{x}}_q) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}_q) \cdot \hat{\nabla} \hat{\phi}_j(\hat{\mathbf{x}}_q) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}_q) |\mathbf{J}_K(\hat{\mathbf{x}}_q)| \mathbf{w}(\hat{\mathbf{x}}_q),
\end{aligned} \tag{3.2.68}$$

while for the force vector, we have

$$\begin{aligned}
h_i &= \sum_{K \in \Omega_h} \int_K \phi_i(\mathbf{x}) dK = \sum_{K \in \Omega_h} \int_{\hat{K}} \hat{\phi}_i(\hat{\mathbf{x}}) |\mathbf{J}_K(\hat{\mathbf{x}})| d\hat{K} \\
&= \sum_{K \in \Omega_h} \sum_q \hat{\phi}_i(\hat{\mathbf{x}}_q) |\mathbf{J}_K(\hat{\mathbf{x}}_q)| \mathbf{w}(\hat{\mathbf{x}}_q).
\end{aligned} \tag{3.2.69}$$

For the  $\rho_a$ -dependent matrix, we have

$$\begin{aligned}
p_{ij} &= \sum_{K \in \Omega_h} \int_K \frac{\nabla \phi_i(\mathbf{x}) \cdot \nabla \phi_j(\mathbf{x})}{1 + a \sum_{k=1}^{N_h} \rho_k(t) \phi_k(\mathbf{x})} dK \\
&= \sum_{K \in \Omega_h} \int_{\hat{K}} \frac{\hat{\nabla} \hat{\phi}_i(\hat{\mathbf{x}}) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}) \cdot \hat{\nabla} \hat{\phi}_j(\hat{\mathbf{x}}) \mathbf{J}_K^{-1}(\hat{\mathbf{x}})}{1 + a \sum_{k=1}^{N_h} \rho_k(t) \hat{\phi}_k(\hat{\mathbf{x}})} |\mathbf{J}_K(\hat{\mathbf{x}})| d\hat{K} \\
&= \sum_{K \in \Omega_h} \sum_q \frac{\hat{\nabla} \hat{\phi}_i(\hat{\mathbf{x}}_q) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}_q) \cdot \hat{\nabla} \hat{\phi}_j(\hat{\mathbf{x}}_q) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}_q)}{1 + a \sum_{k=1}^{N_h} \rho_k(t) \hat{\phi}_k(\hat{\mathbf{x}}_q)} |\mathbf{J}_K(\hat{\mathbf{x}}_q)| \mathbf{w}(\hat{\mathbf{x}}_q),
\end{aligned} \tag{3.2.70}$$

and the nonlinear vector by

$$\begin{aligned}
l_i &= \sum_{K \in \Omega_h} \int_K \frac{\phi_i(\mathbf{x}) \left( \sum_{k=1}^{N_h} \rho_k(t) \phi_k(\mathbf{x}) \right)^2}{1 + \left( \sum_{k=1}^{N_h} \rho_k(t) \phi_k(\mathbf{x}) \right)^2} dK \\
&= \sum_{K \in \Omega_h} \int_{\hat{K}} \frac{\hat{\phi}_i(\hat{\mathbf{x}}) \left( \sum_{k=1}^{N_h} \rho_k(t) \hat{\phi}_k(\hat{\mathbf{x}}) \right)^2}{1 + \left( \sum_{k=1}^{N_h} \rho_k(t) \hat{\phi}_k(\hat{\mathbf{x}}) \right)^2} |\mathbf{J}_K(\hat{\mathbf{x}})| d\hat{K} \\
&= \sum_{K \in \Omega_h} \sum_q \frac{\hat{\phi}_i(\hat{\mathbf{x}}_q) \left( \sum_{k=1}^{N_h} \rho_k(t) \hat{\phi}_k(\hat{\mathbf{x}}_q) \right)^2}{1 + \left( \sum_{k=1}^{N_h} \rho_k(t) \hat{\phi}_k(\hat{\mathbf{x}}_q) \right)^2} |\mathbf{J}_K(\hat{\mathbf{x}}_q)| \mathbf{w}(\hat{\mathbf{x}}_q).
\end{aligned} \tag{3.2.71}$$

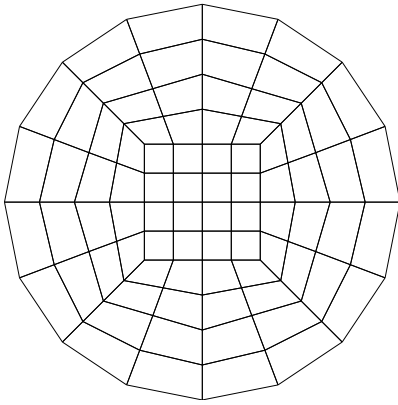
To implement the discretisation, we use **deal.II** library [Bangerth et al. \(2007\)](#). We note that both the mass and stiffness matrices  $\mathbf{M}$  and  $\mathbf{A}$  respectively are large, sparse, symmetric and positive definite while  $\mathbf{P}$  is sparse and non-symmetric. A sparse matrix has very few nonzero entries. Iterative methods are the best suited for such systems [Saad \(2003\)](#). Here, we will use iterative methods that use pre-conditioners. The rate of convergence of an iterative method depends greatly on the spectrum of the matrices, that is, the rate of convergence depends on the condition number of the matrices [Saad \(2003\)](#). A pre-conditioner gives rise to a more favourable spectrum of the resultant matrices [Barrett et al. \(1994\)](#); [Madzvamuse \(2000\)](#). Since the matrices  $\mathbf{M}$  and  $\mathbf{A}$  are both symmetric and positive definite, the system (3.2.66) can be solved using a preconditioned conjugate gradient method (PCG) [Saad \(2003\)](#); [Barrett et al. \(1994\)](#) with a diagonal pre-conditioner. The system (3.2.65) will be solved using a generalised minimal residual method (GMRES) with a diagonal preconditioner as illustrated in [Freund et al. \(1992\)](#); [Saad \(2003\)](#).

### Numerical simulations

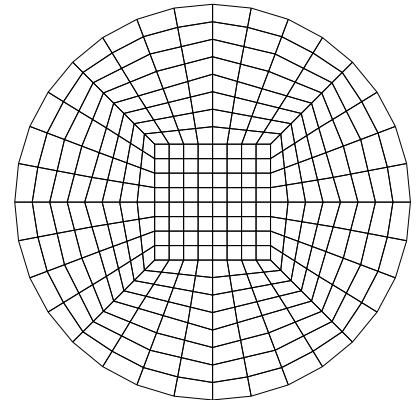
Next, we present some results for the biochemical model. The implementation of the model equations (3.2.65) and (3.2.66) was carried out by extending the library in **deal.II**. We use the unit disk to denote the domain and seek solutions of the  $\rho_a$  and  $\rho_m$  inside this unit disk. We start by illustrating triangulations of the unit disk by carrying out global mesh refinements. We note that as the number of global refinements increase, the triangulations converge to the unit disk as shown in Figure 3.9. The initial data for both variables is chosen to be random perturbation about  $\rho_m = \rho_a = 1.0$  in the entire domain. The simulations were allowed to run until time  $t = 6$  which was long enough for convergence to take place. We have chosen the second order semi-implicit backward differentiation formula, 2-SBDF, scheme with  $\tau = 2 \times 10^{-3}$ ,  $\tau = 10^{-3}$  and  $\tau = 5 \times 10^{-4}$  and compared the  $L_2$ -norms  $\|\frac{\rho_a^{n+1} - \rho_a^n}{\tau}\|$  for the solutions of  $\rho_a$  variable as shown in Figure 3.13. Mesh refinements is also performed and comparison for  $L_2$ -norms done. We note that solutions for both the  $\rho_a$  and  $\rho_m$  variables converge to homogeneous steady state as shown in Figure 3.10 and Figure 3.11. This is expected because of the nature of our model equations on stationary domain. Our reaction kinetics for actin is only a function of F-actin  $\rho_a$  and actin monomers  $\rho_a^{cyl}$  and is independent of  $\rho_m$ . That is, actin only converts from active state to inactive state and vice-versa. Since the total amount of actin is conserved,  $\rho_a$  and  $\rho_a^{cyl}$  concentrations will reach equilibrium states. We plot the conservation of mass for actin in Figure 3.12. We note that  $\rho_a$  and  $\rho_m$  equations are coupled in the diffusion of  $\rho_m$  in such a way that diffusion of  $\rho_m$  variable is reduced when the concentration of  $\rho_a$  is high and vice-versa. This implies that solutions for the  $\rho_m$  variable will converge to a homogeneous steady state when the  $\rho_a$  solution has converged. We also plot the graph of the diffusion of  $\rho_m$  variable versus time in Figure 3.12.

### Discussion of the numerical results

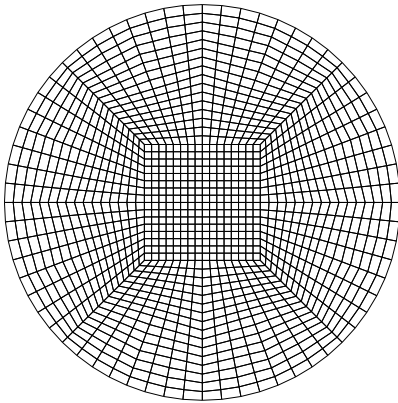
We summarise this section by discussing the results for the biochemical model for F-actin and myosin II. We have developed a finite element method by deriving the weak formulation of the



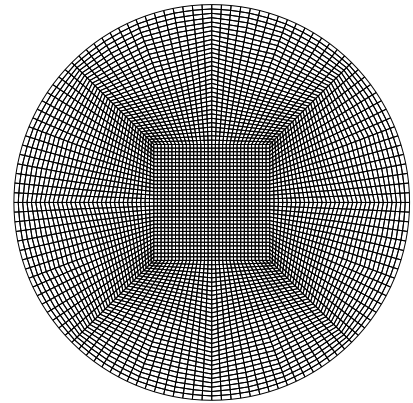
(a)



(b)



(c)



(d)

Figure 3.9: Triangulation of the unit disk using quadrilateral elements after (a) two global mesh refinements, (b) three global mesh refinements, (c) four global mesh refinements and (d) five global mesh refinements showing convergence to the unit disk.

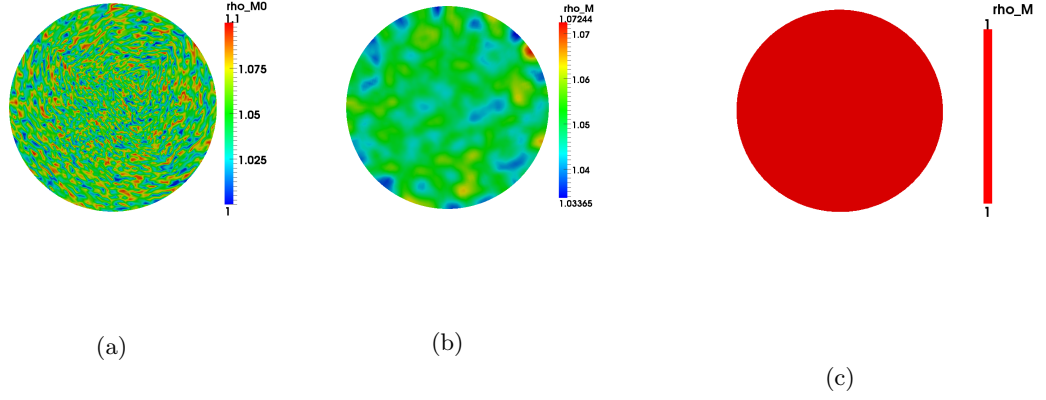


Figure 3.10: Solutions for the  $\rho_m$  variable on a stationary unit disk using  $\tau = 2 \times 10^{-3}$ . Blue signifies the lowest values while red the highest values. (a) Initial condition for myosin II (b)  $\rho_m$  solution at time  $t = 0.8$  and (c)  $\rho_m$  solution at time  $t = 3$  showing convergence to a homogeneous steady state.

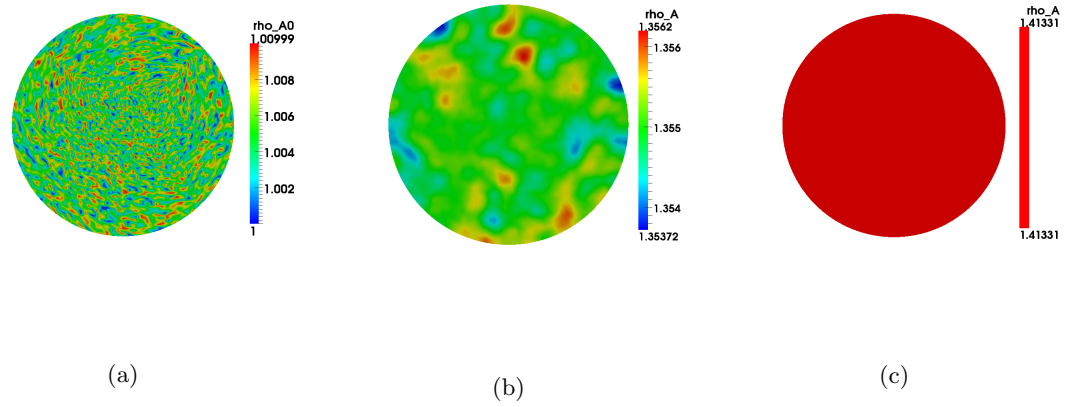


Figure 3.11: Solutions for the  $\rho_a$  variable on a stationary unit disk using  $\tau = 2 \times 10^{-3}$ . Blue signifies the lowest values while red the highest values. (a) Initial condition for F-actin (b)  $\rho_a$  solution at time  $t = 0.8$  and (c)  $\rho_a$  solution at time  $t = 3$  showing convergence to a homogeneous steady state.

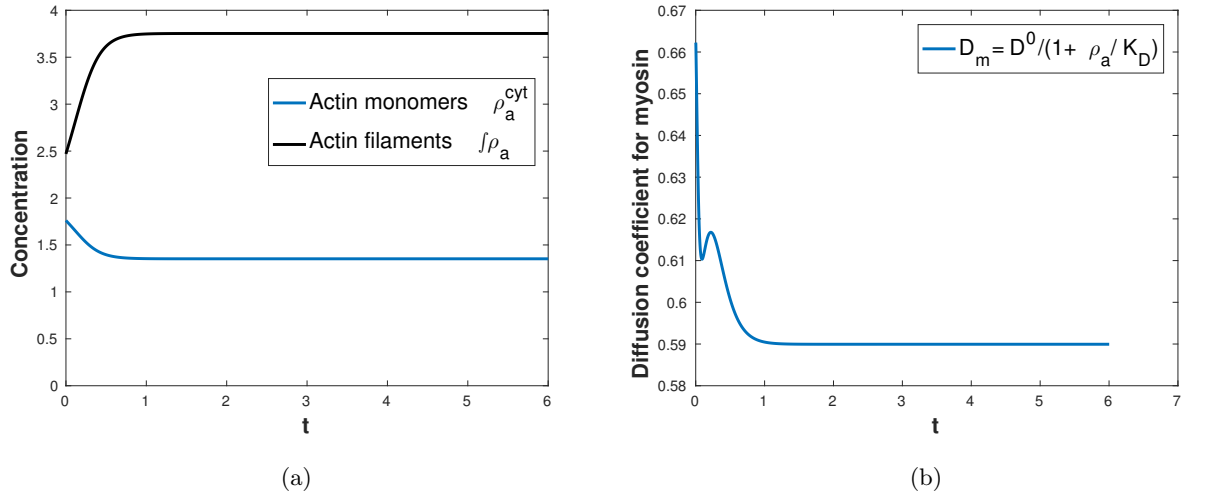


Figure 3.12: (a) Conservation of mass for actin and (b) diffusion coefficient for myosin II. We used time-step  $\tau = 2 \times 10^{-3}$  and mesh size  $h = 0.055126$ .

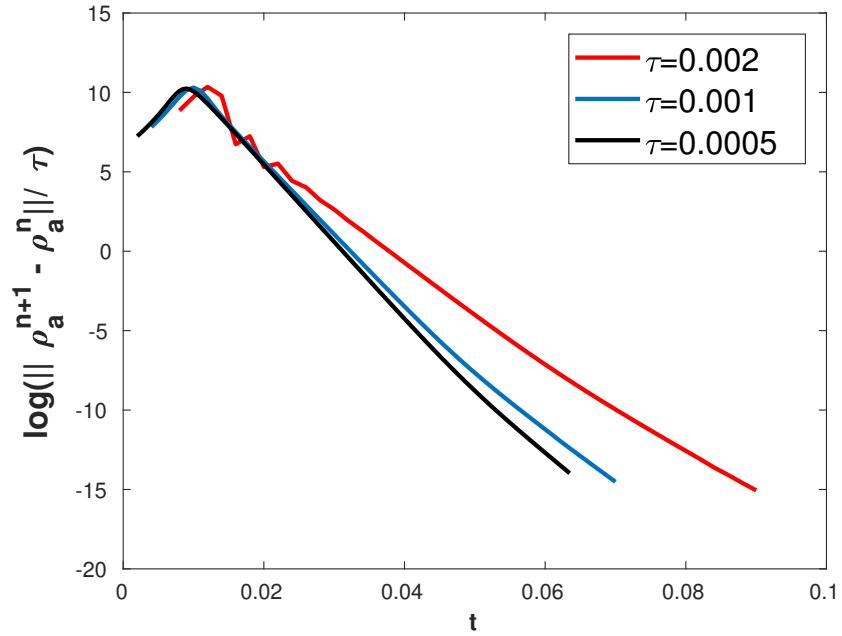


Figure 3.13: Convergence of the  $\rho_a$  variable with different time steps  $\tau$ .

reaction-diffusion system, then discretised the space using finite elements. The resulting semi-discrete equations were then discretised in time using the second order semi-implicit backward differentiation formula to obtain a fully discrete system. We then carried out implementation of the discrete equations using deal.II software. The integrals for the global matrices and vectors were discretised in deal.II using quadrature formula. The resulting linear systems were solved using a preconditioned CG and GMRES methods to obtain solutions for F-actin and myosin II variables.

Initial conditions for the model was assumed to be random perturbation about  $\rho_a = \rho_m = 1$ . We ran simulations until homogeneous steady states was reached. We expect the solution for the two variables to converge to homogeneous steady states with time. This is because of the nature of the biochemical model. The reaction kinetics for F-actin is only a function of F-actin ( $\rho_a$ ) and actin monomers ( $\rho_a^{cyl}$ ) and is independent of  $\rho_m$ . That is, actin only converts from its active state to inactive state and vice-versa. Since the total amount of actin is conserved,  $\rho_a$  and  $\rho_a^{cyl}$  concentrations will reach equilibrium states. Also, the reaction-diffusion equations are only coupled at the diffusion term for  $\rho_m$  in such a way that diffusion of  $\rho_m$  variable is reduced when the concentration of  $\rho_a$  is high and vice-versa. This implies that solutions for the  $\rho_m$  variable will converge to a homogeneous steady state when the  $\rho_a$  solution converges.

We now turn attention to the full viscous model for cell migration which couples reaction-advection-diffusion equations to a force balance equation.

### The non-dimensionalised viscous model

In the previous section, attention was made to solve the biochemical model for F-actin and myosin II. Here, we aim to solve the non-dimensionalised viscous model for cell migration that couples reaction-advection-diffusion equations of F-actin and myosin II to a force balance equation on an evolving domain  $\Omega_t$ . We start by restating the model on  $\Omega_t$ . The non-dimensional parameter values are displayed in Table 3.3.

$$\left\{ \begin{array}{l} \frac{\partial \rho_m}{\partial t} + \nabla \cdot (\rho_m \beta) = \nabla \cdot \left( \frac{1}{1+a\rho_a} \nabla \rho_m \right), \quad \mathbf{x} \in \Omega_t, \quad t \in (0, T], \\ \frac{\partial \rho_a}{\partial t} + \nabla \cdot (\rho_a \beta) = d\Delta \rho_a + \left( \frac{k_3 \rho_a^2}{1+\rho_a^2} + k_4 \right) \rho_a^{cyl} - e\rho_a, \quad \mathbf{x} \in \Omega_t, \quad t \in (0, T], \\ \nabla \cdot (\sigma_\nu(\mathbf{x}, t) + \sigma_{myo}(\mathbf{x}, t) + \sigma_{poly}(\mathbf{x}, t)) = \mathbf{0}, \quad \mathbf{x} \in \Omega_t, \quad t \in (0, T], \\ \rho_a(\mathbf{x}, 0) = \rho_a^0(\mathbf{x}), \quad \rho_m(\mathbf{x}, 0) = \rho_m^0(\mathbf{x}), \quad \mathbf{x} \in \Omega_t, \quad t = 0, \\ \sigma_\nu \cdot \mathbf{n} = \mathbf{0}, \quad \mathbf{x} \in \partial\Omega_t, \quad t \in (0, T], \\ \frac{\partial \rho_a}{\partial \mathbf{n}} = \frac{\partial \rho_m}{\partial \mathbf{n}} = 0, \quad \mathbf{x} \in \partial\Omega_t, \quad t \in (0, T], \end{array} \right. \quad (3.2.72)$$

with

$$\rho_a^{cyl}(t) = \frac{\rho_a^{tot} - \int_{\Omega_t} \rho_a d\Omega_t}{\int_{\Omega_t} d\Omega_t}.$$



Parameters	Description	Value
$a$	Corresponds to diffusion threshold for myosin II	2.0
$k_3$	Corresponds to actin polymerisation rate due to positive feedback	500.0
$k_4$	Corresponds to base polymerisation rate	5.0
$b$	Coefficient for advection	500.0
$d$	Diffusion ratio	0.4
$e$	F-actin depolymerisation rate	500.0
$\eta_1$	Contraction coefficient rate	$\frac{1}{50}$
$\eta_2$	Protrusion coefficient rate	$\frac{56}{500}$
$\rho_a^{tot}$	Total amount of actin	8.0

Table 3.3: Non-dimensional parameters for the non-dimensionalised viscous model

The terms  $\sigma_\nu(\mathbf{x}, t)$ ,  $\sigma_{myo}(\mathbf{x}, t)$  and  $\sigma_{poly}(\mathbf{x}, t)$  are the viscous, myosin II driven and F-actin generated stresses respectively and are given by

$$\begin{cases} \sigma_\nu(\mathbf{x}, t) = \nabla \beta(\mathbf{x}, t) + (\nabla \beta(\mathbf{x}, t))^T, \\ \sigma_{myo}(\mathbf{x}, t) = \eta_1 \rho_m(\mathbf{x}, t) \mathbf{I}, \quad \eta_1 \in \mathbb{R}^+, \\ \sigma_{poly}(\mathbf{x}, t) = -\eta_2 \rho_a(\mathbf{x}, t) \delta(l) \mathbf{I}, \quad \eta_2 \in \mathbb{R}^+, \end{cases} \quad (3.2.73)$$

where  $\beta(\mathbf{x}, t)$  is the actin network flow velocity. The protrusive stress  $\sigma_{poly}(\mathbf{x}, t)$  is assumed to be confined to the cell periphery because F-actin are oriented in such a way that their barbed ends, where the new actin monomers are added are close to the cell membrane and point outwards. The dot product  $\sigma_\nu \cdot \mathbf{n} = \mathbf{0}$  represents the stress-free boundary condition. The myosin II driven contractile stress,  $\sigma_{myo}(\mathbf{x}, t)$ , is assumed to be isotropic. The diffusion for myosin II,  $D_m(\rho_a)$ , is assumed to depend on F-actin density. We will use the unit disk  $\Omega_0$  to represent the cell at initial time with radius  $r = 1$ . Since the polymerisation force is assumed to work only in the periphery of the cell, we prescribe some region of the disk where this force acts. We let this force act only in the region where the radius is  $r > 0.8$ . For subsequent time, we assume that there exists a family of bijective functions that map the point  $\boldsymbol{\eta} = (\eta_x, \eta_y)$  of the initial domain to point  $\mathbf{x} = (x, y)$  of the current domain  $\Omega_t$ . Consider the mapping  $l : \Omega_t \times (0, T] \rightarrow \mathbb{R}$  and its corresponding mapping  $\hat{l} : \Omega_0 \times (0, T] \rightarrow [0, 1]$  on the initial domain  $\Omega_0$  where  $\hat{l}(\boldsymbol{\eta}, t)$  represents the distance from the centre of  $\Omega_0$  to the point  $\boldsymbol{\eta}$  with  $l(\mathbf{x}(\boldsymbol{\eta}, t), t) = \hat{l}(\boldsymbol{\eta}, t)$ . To denote the region where the polymerisation force act, we define a delta function  $\delta(l)$  such that

$$\delta(l) = \begin{cases} 1 & \text{if the point } (\mathbf{x}, t) \text{ with } l(\mathbf{x}(\boldsymbol{\eta}, t), t) = \hat{l}(\boldsymbol{\eta}, t) \\ & \text{is such that the distance } \sqrt{\eta_x^2 + \eta_y^2} > 0.8 \text{ in the initial domain,} \\ 0 & \text{elsewhere.} \end{cases}$$

We proceed as we have done before with the other models, that is, derive the weak formulations and discretise the domain and the time using finite elements and finite differences respectively.

## Derivation of the weak formulation

### Weak formulation of the reaction-advection-diffusion equations

Reaction-advection-diffusion equations for myosin II and F-actin are respectively given by

$$\begin{cases} \frac{\partial \rho_m}{\partial t} + \nabla \cdot (\rho_m \boldsymbol{\beta}) = \nabla \cdot \left( \frac{1}{1+a\rho_a} \nabla \rho_m \right), \\ \frac{\partial \rho_a}{\partial t} + \nabla \cdot (\rho_a \boldsymbol{\beta}) = d\Delta \rho_a + \left( \frac{k_3 \rho_a^2}{1+\rho_a^2} + k_4 \right) \rho_a^{cyl} - e\rho_a, \end{cases} \quad (3.2.74)$$

where  $\rho_m = \rho_m(\mathbf{x}(t), t)$  and  $\rho_a = \rho_a(\mathbf{x}(t), t)$  are the myosin II and F-actin concentrations respectively and  $\boldsymbol{\beta} = \boldsymbol{\beta}(\mathbf{x}(t), t) = (\beta_1(t), \beta_2(t))$  is the flow velocity of the actin network. Here, we note that the reaction kinetics of F-actin only depends on  $\rho_a$  variable and no reaction kinetics for the myosin II equation.

In order to obtain the weak formulation, we will rearrange (3.2.74). We apply product rule for gradient to the advection terms and write the equations as

$$\frac{\partial \rho_m}{\partial t} + \boldsymbol{\beta} \cdot \nabla \rho_m + \rho_m \nabla \cdot \boldsymbol{\beta} = \nabla \cdot \left( \frac{1}{1+a\rho_a} \nabla \rho_m \right), \quad (3.2.75)$$

and

$$\frac{\partial \rho_a}{\partial t} + \boldsymbol{\beta} \cdot \nabla \rho_a + \rho_a \nabla \cdot \boldsymbol{\beta} = d\Delta \rho_a + \left( \frac{k_3 \rho_a^2}{1+\rho_a^2} + k_4 \right) \rho_a^{cyl} - e\rho_a. \quad (3.2.76)$$

We recall that the quantities  $\frac{\partial \rho_m}{\partial t} + \boldsymbol{\beta} \cdot \nabla \rho_m$  and  $\frac{\partial \rho_a}{\partial t} + \boldsymbol{\beta} \cdot \nabla \rho_a$  are called material derivatives of  $\rho_m$  and  $\rho_a$  and we write them as  $\frac{D\rho_m}{Dt} = \frac{\partial \rho_m}{\partial t} + \boldsymbol{\beta} \cdot \nabla \rho_m$  and  $\frac{D\rho_a}{Dt} = \frac{\partial \rho_a}{\partial t} + \boldsymbol{\beta} \cdot \nabla \rho_a$ . Now, using this definition for material derivatives above, we write (3.2.75) and (3.2.76) as

$$\frac{D\rho_m}{Dt} + \rho_m \nabla \cdot \boldsymbol{\beta} = \nabla \cdot \left( \frac{1}{1+a\rho} \nabla \rho_m \right), \quad (3.2.77)$$

and

$$\frac{D\rho_a}{Dt} + \rho_a \nabla \cdot \boldsymbol{\beta} = d\Delta \rho_a + \left( \frac{k_3 \rho_a^2}{1+\rho_a^2} + k_4 \right) \rho_a^{cyl} - e\rho_a. \quad (3.2.78)$$

In order to obtain the weak formulations, we multiply (3.2.77) and (3.2.78) by test functions  $\psi_1(\mathbf{x}, t)$ ,  $\psi_2(\mathbf{x}, t) \in \mathbf{H}^1(\Omega_t)$ ,  $t > 0$  and integrate using Green's formula (2.2.7) in the domain  $\Omega_t$  and use the boundary conditions  $\frac{\partial \rho_a}{\partial \mathbf{n}} = \frac{\partial \rho_m}{\partial \mathbf{n}} = 0$ . This yields

$$\int_{\Omega_t} \left( \psi_1 \frac{D\rho_m}{Dt} + \psi_1 \rho_m \nabla \cdot \boldsymbol{\beta} \right) d\Omega_t = - \int_{\Omega_t} \frac{1}{1+a\rho_a} \nabla \psi_1 \cdot \nabla \rho_m d\Omega_t, \quad (3.2.79)$$

and

$$\begin{aligned} \int_{\Omega_t} \left( \psi_2 \frac{D\rho_a}{Dt} + \psi_2 \rho_a \nabla \cdot \boldsymbol{\beta} \right) d\Omega_t &= - \int_{\Omega_t} d\nabla \psi_2 \cdot \nabla \rho_a d\Omega_t \\ &\quad + \int_{\Omega_t} \psi_2 \left( \left( \frac{k_3 \rho_a^2}{1+\rho_a^2} + k_4 \right) \rho_a^{cyl} - e\rho_a \right) d\Omega_t. \end{aligned} \quad (3.2.80)$$

We further use product rule for the time derivatives in the equations (3.2.79) and (3.2.80) and write

$$\int_{\Omega_t} \left( \frac{D(\psi_1 \rho_m)}{Dt} - \rho_m \frac{D\psi_1}{Dt} + \psi_1 \rho_m \nabla \cdot \boldsymbol{\beta} \right) d\Omega_t = - \int_{\Omega_t} \frac{1}{1+a\rho_a} \nabla \psi_1 \cdot \nabla \rho_m d\Omega_t, \quad (3.2.81)$$

and

$$\begin{aligned} \int_{\Omega_t} \left( \frac{D(\psi_2 \rho_a)}{Dt} - \rho_a \frac{D\psi_2}{Dt} + \psi_2 \rho_a \nabla \cdot \beta \right) d\Omega_t &= - \int_{\Omega_t} d\nabla \psi_2 \cdot \nabla \rho_a d\Omega_t \\ &+ \int_{\Omega_t} \psi_2 \left( \left( \frac{k_3 \rho_a^2}{1 + \rho_a^2} + k_4 \right) \rho_a^{c_{yt}} - e \rho_a \right) d\Omega_t. \end{aligned} \quad (3.2.82)$$

Finally Reynold's transport theorem (2.2.8) gives

$$\frac{d}{dt} \int_{\Omega_t} \psi_1 \rho_m d\Omega_t - \int_{\Omega_t} \rho_m \frac{D\psi_1}{Dt} d\Omega_t = - \int_{\Omega_t} \frac{1}{1 + a \rho_a} \nabla \psi_1 \cdot \nabla \rho_m d\Omega_t, \quad (3.2.83)$$

and

$$\begin{aligned} \frac{d}{dt} \int_{\Omega_t} \psi_2 \rho_a d\Omega_t - \int_{\Omega_t} \rho_a \frac{D\psi_2}{Dt} d\Omega_t &= - \int_{\Omega_t} d\nabla \psi_2 \cdot \nabla \rho_a d\Omega_t \\ &+ \int_{\Omega_t} \psi_2 \left( \left( \frac{k_3 \rho_a^2}{1 + \rho_a^2} + k_4 \right) \rho_a^{c_{yt}} - e \rho_a \right) d\Omega_t. \end{aligned} \quad (3.2.84)$$

The weak formulation of the reaction-advection-diffusion equations therefore reads:

find  $\rho_m(\mathbf{x}(t), t)$ ,  $\rho_a(\mathbf{x}(t), t) \in \mathbf{H}^1(\Omega_t)$ ,  $t > 0$  such that

$$\frac{d}{dt} \int_{\Omega_t} \psi_1 \rho_m d\Omega_t - \int_{\Omega_t} \rho_m \frac{D\psi_1}{Dt} d\Omega_t = - \int_{\Omega_t} \frac{1}{1 + a \rho_a} \nabla \psi_1 \cdot \nabla \rho_m d\Omega_t, \quad (3.2.85)$$

and

$$\begin{aligned} \frac{d}{dt} \int_{\Omega_t} \psi_2 \rho_a d\Omega_t - \int_{\Omega_t} \rho_a \frac{D\psi_2}{Dt} d\Omega_t &= - \int_{\Omega_t} d\nabla \psi_2 \cdot \nabla \rho_a d\Omega_t \\ &+ \int_{\Omega_t} \psi_2 \left( \left( \frac{k_3 \rho_a^2}{1 + \rho_a^2} + k_4 \right) \rho_a^{c_{yt}} - e \rho_a \right) d\Omega_t, \end{aligned} \quad (3.2.86)$$

for all  $\psi_1(\mathbf{x}, t)$ ,  $\psi_2(\mathbf{x}, t) \in \mathbf{H}^1(\Omega_t)$ .

### Weak formulation of the force balance equations

The force balance equation on an evolving domain  $\Omega_t$  representing the cell is given by

$$\begin{cases} \nabla \cdot (\sigma_\nu(\mathbf{x}, t) + \sigma_{myo}(\mathbf{x}, t) + \sigma_{poly}(\mathbf{x}, t)) = \mathbf{0}, & \mathbf{x} \in \Omega_t, \quad t \in (0, T], \\ \sigma_\nu \cdot \mathbf{n} = \mathbf{0}, & \mathbf{x} \in \partial\Omega_t, \quad t \in (0, T], \end{cases} \quad (3.2.87)$$

where  $\sigma_\nu(\mathbf{x}, t)$ ,  $\sigma_{myo}(\mathbf{x}, t)$  and  $\sigma_{poly}(\mathbf{x}, t)$  are the viscous, myosin II driven and F-actin generated stresses respectively and are given by

$$\begin{cases} \sigma_\nu(\mathbf{x}, t) = \nabla \beta(\mathbf{x}, t) + (\nabla \beta(\mathbf{x}, t))^T, \\ \sigma_{myo}(\mathbf{x}, t) = \eta_1 \rho_m(\mathbf{x}, t) \mathbf{I}, \quad \eta_1 \in \mathbb{R}^+, \\ \sigma_{poly}(\mathbf{x}, t) = -\eta_2 \rho_a(\mathbf{x}, t) \delta(l) \mathbf{I}, \quad \eta_2 \in \mathbb{R}^+. \end{cases} \quad (3.2.88)$$

In order to write the weak formulation of the force balance equation above, we first decouple the stresses into  $x$  and  $y$  directions as follows.

$$\begin{cases} \boldsymbol{\sigma}_\nu(\mathbf{x}, t) = \begin{pmatrix} 2\frac{\partial\beta_1}{\partial x} & \left(\frac{\partial\beta_2}{\partial x} + \frac{\partial\beta_1}{\partial y}\right) \\ \left(\frac{\partial\beta_2}{\partial x} + \frac{\partial\beta_1}{\partial y}\right) & 2\frac{\partial\beta_2}{\partial y} \end{pmatrix}, \\ \boldsymbol{\sigma}_{m\mathbf{y}\mathbf{o}}(\mathbf{x}, t) = \begin{pmatrix} \eta_1\rho_m & 0 \\ 0 & \eta_1\rho_m \end{pmatrix}, \\ \boldsymbol{\sigma}_{poly}(\mathbf{x}, t) = \begin{pmatrix} -\eta_2\rho_a\delta(l) & 0 \\ 0 & -\eta_2\rho_a\delta(l) \end{pmatrix}. \end{cases} \quad (3.2.89)$$

The force balance equation in  $x$  and  $y$  directions is therefore

$$\frac{\partial}{\partial x} \left( 2\frac{\partial\beta_1}{\partial x} \right) + \frac{\partial}{\partial y} \left( \frac{\partial\beta_2}{\partial x} + \frac{\partial\beta_1}{\partial y} \right) + \frac{\partial(\eta_1\rho_m)}{\partial x} - \frac{\partial}{\partial x} (\eta_2\rho_a\delta(l)) = 0, \quad (3.2.90)$$

$$\frac{\partial}{\partial x} \left( \frac{\partial\beta_2}{\partial x} + \frac{\partial\beta_1}{\partial y} \right) + \frac{\partial}{\partial y} \left( 2\frac{\partial\beta_2}{\partial y} \right) + \frac{\partial(\eta_1\rho_m)}{\partial y} - \frac{\partial}{\partial y} (\eta_2\rho_a\delta(l)) = 0. \quad (3.2.91)$$

We then multiply the decoupled equations (3.2.90) and (3.2.91) by a test function  $\psi_3 \in \mathbf{H}^1(\Omega_t)$ , use Green's formula to integrate in the domain and apply the stress free boundary condition given. The boundary terms will vanish and the weak formulation thus reads: find  $\beta_1(\mathbf{x}(t), t)$ ,  $\beta_2(\mathbf{x}(t), t) \in \mathbf{H}^1(\Omega_t)$  such that

$$\int_{\Omega_t} \left( 2\frac{\partial\psi_3}{\partial x} \left( \frac{\partial\beta_1}{\partial x} \right) + \frac{\partial\psi_3}{\partial y} \left( \frac{\partial\beta_2}{\partial x} + \frac{\partial\beta_1}{\partial y} \right) \right) d\Omega_t = \int_{\Omega_t} \psi_3 \left( \frac{\partial f}{\partial x} \right) d\Omega_t, \quad (3.2.92)$$

and

$$\int_{\Omega_t} \left( \frac{\partial\psi_3}{\partial x} \left( \frac{\partial\beta_2}{\partial x} + \frac{\partial\beta_1}{\partial y} \right) + 2\frac{\partial\psi_3}{\partial y} \left( \frac{\partial\beta_2}{\partial y} \right) \right) d\Omega_t = \int_{\Omega_t} \psi_3 \left( \frac{\partial f}{\partial y} \right) d\Omega_t, \quad (3.2.93)$$

for all  $\psi_3 \in \mathbf{H}^1(\Omega_t)$ , where

$$f = \delta(l)\eta_2\rho_a - \eta_1\rho_m.$$

We find it convenient to rewrite the right hand sides of equations (3.2.92) and (3.2.93) such that we have the derivatives of the shape function  $\psi_3$  instead of function  $f$  which depends on the solution values. We do so using the gradient formulae (2.2.5) and (2.2.6). This gives the following weak formulation: find  $\beta_1(\mathbf{x}(t), t)$ ,  $\beta_2(\mathbf{x}(t), t) \in \mathbf{H}^1(\Omega_t)$  such that

$$\int_{\Omega_t} \left( 2\frac{\partial\psi_3}{\partial x} \left( \frac{\partial\beta_1}{\partial x} \right) + \frac{\partial\psi_3}{\partial y} \left( \frac{\partial\beta_2}{\partial x} + \frac{\partial\beta_1}{\partial y} \right) \right) d\Omega_t = - \int_{\Omega_t} f \frac{\partial\psi_3}{\partial x} d\Omega_t + \int_{\partial\Omega_t} (n_1 f \psi_3) dS, \quad (3.2.94)$$

and

$$\int_{\Omega_t} \left( \frac{\partial\psi_3}{\partial x} \left( \frac{\partial\beta_2}{\partial x} + \frac{\partial\beta_1}{\partial y} \right) + 2\frac{\partial\psi_3}{\partial y} \left( \frac{\partial\beta_2}{\partial y} \right) \right) d\Omega_t = - \int_{\Omega_t} f \frac{\partial\psi_3}{\partial y} d\Omega_t + \int_{\partial\Omega_t} (n_2 f \psi_3) dS, \quad (3.2.95)$$

for all  $\psi_3 \in \mathbf{H}^1(\Omega_t)$ , where

$$f = \delta(l)\eta_2\rho_a - \eta_1\rho_m,$$

and  $\mathbf{n} = (n_1, n_2)$  is the outward unit normal to the boundary.

### Finite element discretisation

Solutions for the weak formulations are defined in an infinite dimensional space  $\mathbf{H}^1(\Omega_t)$ . The essence of the finite element method is to seek solutions in a finite dimensional space. We start by restating the weak formulations of the coupled problem:

find  $\rho_m(\mathbf{x}(t), t)$ ,  $\rho_a(\mathbf{x}(t), t)$ ,  $\beta_1(\mathbf{x}(t), t)$ ,  $\beta_2(\mathbf{x}(t), t) \in \mathbf{H}^1(\Omega_t)$  such that

$$\frac{d}{dt} \int_{\Omega_t} \psi_1 \rho_m d\Omega_t - \int_{\Omega_t} \rho_m \frac{D\psi_1}{Dt} d\Omega_t = - \int_{\Omega_t} \frac{1}{1 + a\rho_a} \nabla \psi_1 \cdot \nabla \rho_m d\Omega_t, \quad (3.2.96)$$

$$\begin{aligned} \frac{d}{dt} \int_{\Omega_t} \psi_2 \rho_a d\Omega_t - \int_{\Omega_t} \rho_a \frac{D\psi_2}{Dt} d\Omega_t &= - \int_{\Omega_t} d\nabla \psi_2 \cdot \nabla \rho_a d\Omega_t \\ &+ \int_{\Omega_t} \psi_2 \left( \left( \frac{k_3 \rho_a^2}{1 + \rho_a^2} + k_4 \right) \rho_a^{cyl} - e\rho_a \right) d\Omega_t, \end{aligned} \quad (3.2.97)$$

$$\int_{\Omega_t} \left( 2 \frac{\partial \psi_3}{\partial x} \left( \frac{\partial \beta_1}{\partial x} \right) + \frac{\partial \psi_3}{\partial y} \left( \frac{\partial \beta_2}{\partial x} + \frac{\partial \beta_1}{\partial y} \right) \right) d\Omega_t = - \int_{\Omega_t} f \frac{\partial \psi_3}{\partial x} d\Omega_t + \int_{\partial \Omega_t} (n_1 f \psi_3) dS, \quad (3.2.98)$$

and

$$\int_{\Omega_t} \left( \frac{\partial \psi_3}{\partial x} \left( \frac{\partial \beta_2}{\partial x} + \frac{\partial \beta_1}{\partial y} \right) + 2 \frac{\partial \psi_3}{\partial y} \left( \frac{\partial \beta_2}{\partial y} \right) \right) d\Omega_t = - \int_{\Omega_t} f \frac{\partial \psi_3}{\partial y} d\Omega_t + \int_{\partial \Omega_t} (n_2 f \psi_3) dS, \quad (3.2.99)$$

for all  $\{\psi_k\}_{k=1}^3 \in \mathbf{H}^1(\Omega_t)$ , where

$$f = \delta(l) \eta_2 \rho_a - \eta_1 \rho_m.$$

We let  $\Omega_{h,t}$  be the computational domain which is a polyhedral approximation to  $\Omega_t$ . We define  $T_h(t)$  to be a triangulation of  $\Omega_{h,t}$  made up of non-degenerate rectangular elements  $K_i$  such that  $T_h(t) = \bigcup_i K_i$ . We call each  $K_i$  an element of the mesh  $T_h(t)$  where  $h$  is the diameter of the largest element. For the mesh  $T_h(t)$ , we require that it is made up of a finite number of elements and the elements must intersect along a complete edge, or at a vertex or not at all. The space discretisation is carried out using quadrilateral elements and we seek piece-wise linear approximation of the solution. We define the finite element space  $\mathbf{V}_h(t)$  by

$$\mathbf{V}_h(t) = \{v_h(t) \in C^0(\Omega) : v_h(t)|_K \text{ is linear}\} \quad (3.2.100)$$

We will seek solutions of the viscous model in this space. The discretised version of (3.2.96)-(3.2.99) therefore reads: find

$$\rho_m^h(\mathbf{x}(t), t), \rho_a^h(\mathbf{x}(t), t), \beta_1^h(\mathbf{x}(t), t), \beta_2^h(\mathbf{x}(t), t) \in \mathbf{V}_h(t)$$

such that

$$\frac{d}{dt} \int_{\Omega_{h,t}} \psi_1^h \rho_m^h d\Omega_{h,t} - \int_{\Omega_{h,t}} \rho_m^h \frac{D\psi_1^h}{Dt} d\Omega_{h,t} = - \int_{\Omega_{h,t}} \frac{1}{1 + a\rho_a^h} \nabla \psi_1^h \cdot \nabla \rho_m^h d\Omega_{h,t}, \quad (3.2.101)$$

$$\begin{aligned} \frac{d}{dt} \int_{\Omega_{h,t}} \psi_2^h \rho_a^h d\Omega_{h,t} - \int_{\Omega_{h,t}} \rho_a^h \frac{D\psi_2^h}{Dt} d\Omega_{h,t} &= - \int_{\Omega_{h,t}} d\nabla \psi_2^h \cdot \nabla \rho_a^h d\Omega_{h,t} \\ &+ \int_{\Omega_{h,t}} \psi_2^h \left( \left( \frac{k_3 (\rho_a^h)^2}{1 + (\rho_a^h)^2} + k_4 \right) \rho_a^{cyl,h} - e\rho_a^h \right) d\Omega_{h,t}, \end{aligned} \quad (3.2.102)$$

$$\begin{aligned} \int_{\Omega_{h,t}} \left( 2 \frac{\partial \psi_3^h}{\partial x} \left( \frac{\partial \beta_1^h}{\partial x} \right) + \frac{\partial \psi_3^h}{\partial y} \left( \frac{\partial \beta_2^h}{\partial x} + \frac{\partial \beta_1^h}{\partial y} \right) \right) d\Omega_{h,t} = & - \int_{\Omega_{h,t}} f^h \frac{\partial \psi_3^h}{\partial x} d\Omega_{h,t} \\ & + \int_{\partial\Omega_{h,t}} n_1 f^h \psi_3^h dS, \end{aligned} \quad (3.2.103)$$

and

$$\begin{aligned} \int_{\Omega_{h,t}} \left( \frac{\partial \psi_3^h}{\partial x} \left( \frac{\partial \beta_2^h}{\partial x} + \frac{\partial \beta_1^h}{\partial y} \right) + 2 \frac{\partial \psi_3^h}{\partial y} \left( \frac{\partial \beta_2^h}{\partial y} \right) \right) d\Omega_{h,t} = & - \int_{\Omega_{h,t}} f^h \frac{\partial \psi_3^h}{\partial y} d\Omega_{h,t} \\ & + \int_{\partial\Omega_{h,t}} n_2 f^h \psi_3^h dS, \end{aligned} \quad (3.2.104)$$

for all  $\{\psi_k^h\}_{k=1}^3 \in \mathbf{V}_h(t)$ , where

$$f^h = \delta(l) \eta_2 \rho_a^h - \eta_1 \rho_m^h. \quad (3.2.105)$$

We define a basis function for the space  $\mathbf{V}_h(t)$  by  $\phi_i(\mathbf{x}, t) \in \mathbf{V}_h(t)$  for  $i = 1, 2, \dots, N_h$  such that

$$\phi_i(\mathbf{x}_j, t) = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{if } i \neq j, \end{cases} \quad (3.2.106)$$

where  $\mathbf{x}_j(t)$  is the  $j$ th nodal point of the mesh and  $N_h$  is the total number of degrees of freedom of the nodes. We seek finite element approximations of the form

$$\begin{cases} \rho_m^h(\mathbf{x}, t) = \sum_{j=1}^{N_h} \omega_j(t) \phi_j(\mathbf{x}, t), \\ \rho_a^h(\mathbf{x}, t) = \sum_{j=1}^{N_h} \rho_j(t) \phi_j(\mathbf{x}, t), \\ \beta_1^h(\mathbf{x}, t) = \sum_{j=1}^{N_h} U_j(t) \phi_j(\mathbf{x}, t), \\ \beta_2^h(\mathbf{x}, t) = \sum_{j=1}^{N_h} V_j(t) \phi_j(\mathbf{x}, t). \end{cases} \quad (3.2.107)$$

We note that the shape function  $\phi_i$  is now a function of time  $t$ . We will make use of the following Lemma:

**Lemma:** Transport property of the basis functions: The finite element space on the discretised domain is a space of continuous piece-wise linear functions whose nodal basis functions have the remarkable property

$$\frac{D\phi_i(\mathbf{x}, t)}{Dt} \Big|_K = 0 \quad (3.2.108)$$

on element  $K$  for all  $\phi_i$  where the derivative denotes the material derivative [Dziuk and Elliott \(2007\)](#).

**Proof:** Each quadrilateral  $K(t) \in \Omega_{h,t}$  with vertices  $\mathbf{x}_k$ ,  $k = 1, 2, 3, 4$  can be parametrised using co-ordinates in the canonical quadrilateral  $\hat{K}$ . If we let the co-ordinates in the canonical element be given by  $(\xi, \eta)$  and the shape functions in this canonical element be  $\hat{\phi}_k$ , then we can write any point on  $K(t)$  by

$$\mathbf{x}(t) = \sum_{k=1}^4 \mathbf{x}_k(t) \hat{\phi}_k(\xi, \eta). \quad (3.2.109)$$

Also, each  $\phi_i(\mathbf{x}, t)$ ,  $i = 1, 2, 3, 4$  can be transformed as  $\phi_i(\mathbf{x}, t) = \hat{\phi}_i(f(\xi, \eta))$ . Now we have

$$\frac{D\phi_i(\mathbf{x}, t)}{Dt} \Big|_K = \frac{D\hat{\phi}_i(f(\xi, \eta))}{Dt} \Big|_{\hat{K}} = \frac{\partial \hat{\phi}_i(f(\xi, \eta))}{\partial t} \Big|_{\hat{K}} + \beta \cdot \nabla \hat{\phi}_i(f(\xi, \eta)) \Big|_{\hat{K}}. \quad (3.2.110)$$

and have

$$\frac{D\phi_i(\mathbf{x}, t)}{Dt}|_K = \frac{D\hat{\phi}_i(f(\xi, \eta))}{Dt}|_{\hat{K}} = 0, \quad (3.2.111)$$

which completes the proof

### Semi-discrete equations for the reaction-advection-diffusion equations

In equations (3.2.101) and (3.2.102), we substitute  $\{\psi_l^h(\mathbf{x}, t)\}_{l=1}^2$  by  $\phi_i(\mathbf{x}, t)$ ,  $i = 1, 2, \dots, N_h$  and  $\rho_m^h, \rho_a^h$  by their corresponding finite element approximations and use (3.2.108). This gives

$$\frac{d}{dt} \left( \sum_{j=1}^{N_h} \int_{\Omega_{h,t}} \phi_i(\mathbf{x}, t) \cdot \phi_j(\mathbf{x}, t) \omega_j(t) d\Omega_{h,t} \right) = - \sum_{j=1}^{N_h} \int_{\Omega_{h,t}} \frac{\nabla \phi_i(\mathbf{x}, t) \cdot \nabla \phi_j(\mathbf{x}, t) \omega_j(t)}{1 + a \sum_{k=1}^{N_h} \rho_k(t) \phi_k(\mathbf{x}, t)} d\Omega_{h,t},$$

and

$$\begin{aligned} \frac{d}{dt} \left( \sum_{j=1}^{N_h} \int_{\Omega_{h,t}} \phi_i(\mathbf{x}, t) \cdot \phi_j(\mathbf{x}, t) \rho_j(t) d\Omega_{h,t} \right) = & -d \sum_{j=1}^{N_h} \int_{\Omega_{h,t}} \nabla \phi_i(\mathbf{x}, t) \cdot \nabla \phi_j(\mathbf{x}, t) \rho_j(t) d\Omega_{h,t} \\ & - e \sum_{j=1}^{N_h} \int_{\Omega_{h,t}} \phi_i(\mathbf{x}, t) \cdot \phi_j(\mathbf{x}, t) \rho_j(t) d\Omega_{h,t} \\ & + k_4 a_{mon}(t) \int_{\Omega_{h,t}} \phi_i(\mathbf{x}, t) d\Omega_{h,t} \\ & + k_3 a_{mon}(t) \int_{\Omega_{h,t}} \frac{\phi_i(\mathbf{x}, t) \left( \sum_{j=1}^{N_h} \rho_j(t) \phi_j(\mathbf{x}, t) \right)^2}{1 + \left( \sum_{j=1}^{N_h} \rho_j(t) \phi_j(\mathbf{x}, t) \right)^2} d\Omega_{h,t}, \end{aligned}$$

respectively, for all  $i = 1, 2, \dots, N_h$ . The parameter  $a_{mon}(t)$  represents the well mixed actin monomers concentration at time  $t$ . Now, integrating over  $\Omega_{h,t}$  yields the semi-discrete equations for the reaction-advection-diffusion equations as

$$\frac{d}{dt} (\mathbf{M}(t) \boldsymbol{\omega}(t)) = -\mathbf{S}(\rho(t)) \boldsymbol{\omega}(t), \quad (3.2.112)$$

and

$$\frac{d}{dt} (\mathbf{M}(t) \boldsymbol{\rho}(t)) = -(d\mathbf{K}(t) + e\mathbf{M}(t)) \boldsymbol{\rho}(t) + a_{mon}(t) k_4 \mathbf{H}(t) + k_3 a_{mon}(t) \mathbf{L}(\rho(t)), \quad (3.2.113)$$

where  $\boldsymbol{\omega}(t) = (\omega_1(t), \omega_2(t), \dots, \omega_{N_h}(t))^T$  and  $\boldsymbol{\rho}(t) = (\rho_1(t), \rho_2(t), \dots, \rho_{N_h}(t))^T$  are the solution vectors and  $a_{mon}(t)$  is actin monomers concentration at time  $t$ .  $\mathbf{M}(t)$  is the time-dependent global mass matrix,  $\mathbf{K}(t)$  is the time-dependent global stiffness matrix,  $\mathbf{H}(t)$  is the time-dependent global force vector and  $\mathbf{S}(\rho(t))$  and  $\mathbf{L}(\rho(t))$  are the time-dependent matrix and vector respectively which

are functions of the solution vectors. These are given by

$$\begin{aligned}
\mathbf{M}(t) &= \{m_{ij}(t)\} : m_{ij}(t) = \int_{\Omega_{h,t}} \phi_i(\mathbf{x}, t) \cdot \phi_j(\mathbf{x}, t) d\Omega_{h,t}, \\
\mathbf{K}(t) &= \{k_{ij}(t)\} : k_{ij}(t) = \int_{\Omega_{h,t}} \nabla \phi_i(\mathbf{x}, t) \cdot \nabla \phi_j(\mathbf{x}, t) d\Omega_{h,t}, \\
\mathbf{S}(\rho(t)) &= \{s_{ij}(t)\} : s_{ij}(t) = \int_{\Omega_{h,t}} \frac{\nabla \phi_i(\mathbf{x}, t) \cdot \nabla \phi_j(\mathbf{x}, t)}{1 + a \sum_{k=1}^{N_h} \rho_k(t) \phi_k(\mathbf{x}, t)} d\Omega_{h,t}, \\
\mathbf{L}(\rho(t)) &= \{l_i(t)\} : l_i(t) = \int_{\Omega_{h,t}} \frac{\phi_i(\mathbf{x}, t) \left( \sum_{k=1}^{N_h} \rho_k(t) \phi_k(\mathbf{x}, t) \right)^2}{1 + \left( \sum_{k=1}^{N_h} \rho_k(t) \phi_k(\mathbf{x}, t) \right)^2} d\Omega_{h,t}, \\
\mathbf{H}(t) &= \{h_i(t)\} : h_i(t) = \int_{\Omega_{h,t}} \phi_i(\mathbf{x}, t) d\Omega_{h,t}.
\end{aligned}$$

### Semi-discrete equations for the force balance equation

In (3.2.103) and (3.2.104), we substitute  $\psi_3^h(\mathbf{x}, t)$  by  $\phi_i(\mathbf{x}(t), t)$ ,  $i = 1, 2, \dots, N_h$  and  $\beta_1^h, \beta_2^h, \rho_m^h, \rho_a^h$  by their corresponding finite element approximations and integrate over  $\Omega_{h,t}$ . The semi-discrete equation in  $x$  direction will be

$$\begin{aligned}
&\sum_{j=1}^{N_h} \int_{\Omega_{h,t}} \left( 2 \frac{\partial \phi_i}{\partial x} \left( \frac{\partial \phi_j}{\partial x} U_j(t) \right) + \frac{\partial \phi_i}{\partial y} \left( \frac{\partial \phi_j}{\partial x} V_j(t) + \frac{\partial \phi_j}{\partial y} U_j(t) \right) \right) d\Omega_{h,t} = \\
&- \int_{\Omega_{h,t}} \left( f^h(\mathbf{x}(t)) \frac{\partial \phi_i}{\partial x} \right) d\Omega_{h,t} + \int_{\partial \Omega_{h,t}} (n_1 f^h(\mathbf{x}(t)) \phi_i) dS,
\end{aligned} \tag{3.2.114}$$

for all  $i = 1, 2, \dots, N_h$  and can be rearranged as

$$\begin{aligned}
&\sum_{j=1}^{N_h} \int_{\Omega_{h,t}} \left( 2 \frac{\partial \phi_i}{\partial x} \frac{\partial \phi_j}{\partial x} + \frac{\partial \phi_i}{\partial y} \frac{\partial \phi_j}{\partial y} \right) U_j(t) d\Omega_{h,t} + \sum_{j=1}^{N_h} \int_{\Omega_{h,t}} \frac{\partial \phi_i}{\partial y} \frac{\partial \phi_j}{\partial x} V_j(t) d\Omega_{h,t} = \\
&- \int_{\Omega_{h,t}} \left( f^h(\mathbf{x}(t)) \frac{\partial \phi_i}{\partial x} \right) d\Omega_{h,t} + \int_{\partial \Omega_{h,t}} (n_1 f^h(\mathbf{x}(t)) \phi_i) dS,
\end{aligned} \tag{3.2.115}$$

for all  $i = 1, 2, \dots, N_h$ . We note that (3.2.115) can be written as

$$\begin{aligned}
&a_{11}(t)U_1(t) + a_{12}(t)U_2(t) + \dots + a_{1N_h}(t)U_{N_h}(t) + b_{11}(t)V_1(t) + \dots + b_{1N_h}(t)V_{N_h}(t) = F_1^1(t), \\
&a_{21}(t)U_1(t) + a_{22}(t)U_2(t) + \dots + a_{2N_h}(t)U_{N_h}(t) + b_{21}(t)V_1(t) + \dots + b_{2N_h}(t)V_{N_h}(t) = F_2^1(t), \\
&\vdots \\
&a_{N_h 1}(t)U_1(t) + a_{N_h 2}(t)U_2(t) + \dots + a_{N_h N_h}(t)U_{N_h}(t) + b_{N_h 1}(t)V_1(t) + \dots + b_{N_h N_h}(t)V_{N_h}(t) = F_{N_h}^1(t),
\end{aligned}$$

where  $a_{ij}(t)$ ,  $b_{ij}(t)$  and  $F_i^1(t)$  are integrals over  $\Omega_{h,t}$  which we will describe shortly. This means we can split the left hand side of the above system of equations into two parts: one with  $U_j(t)$  and the other with  $V_j(t)$ .



Similarly, the Y direction of the force balance equation is

$$\begin{aligned} & \sum_{j=1}^{N_h} \int_{\Omega_{h,t}} \frac{\partial \phi_i}{\partial x} \frac{\partial \phi_j}{\partial y} U_j(t) d\Omega_{h,t} + \sum_{j=1}^{N_h} \int_{\Omega_{h,t}} \left( \frac{\partial \phi_i}{\partial x} \frac{\partial \phi_j}{\partial x} + 2 \frac{\partial \phi_i}{\partial y} \frac{\partial \phi_j}{\partial y} \right) V_j(t) d\Omega_{h,t} = \\ & - \int_{\Omega_{h,t}} \left( f^h(\mathbf{x}(t)) \frac{\partial \phi_i}{\partial y} \right) d\Omega_{h,t} + \int_{\partial \Omega_{h,t}} (n_2 f^h(\mathbf{x}(t)) \phi_i) dS, \end{aligned} \quad (3.2.116)$$

for all  $i = 1, 2, \dots, N_h$ , and expanded as

$$\begin{aligned} c_{11}(t)U_1(t) + c_{12}(t)U_2(t) + \dots + c_{1N_h}(t)U_{N_h}(t) + d_{11}(t)V_1(t) + \dots + d_{1N_h}(t)V_{N_h}(t) &= F_1^2(t), \\ c_{21}(t)U_1(t) + c_{22}(t)U_2(t) + \dots + c_{2N_h}(t)U_{N_h}(t) + d_{21}(t)V_1(t) + \dots + d_{2N_h}(t)V_{N_h}(t) &= F_2^2(t), \\ &\vdots \\ c_{N_h 1}(t)U_1(t) + c_{N_h 2}(t)U_2(t) + \dots + c_{N_h N_h}(t)U_{N_h}(t) + d_{N_h 1}(t)V_1(t) + \dots + d_{N_h N_h}(t)V_{N_h}(t) &= F_{N_h}^2(t), \end{aligned}$$

with  $c_{ij}(t)$ ,  $d_{ij}(t)$  and  $F_i^2(t)$  integrals over  $\Omega_{h,t}$ . These systems of equations can be written more compactly in matrix-vector form as follows

$$\begin{pmatrix} a_{11} & a_{12} & \dots & a_{1N_h} & b_{11} & b_{12} & \dots & b_{1N_h} \\ a_{21} & a_{22} & \dots & a_{2N_h} & b_{21} & b_{22} & \dots & b_{2N_h} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{N_h 1} & a_{N_h 2} & \dots & a_{N_h N_h} & b_{N_h 1} & b_{N_h 2} & \dots & b_{N_h N_h} \\ c_{11} & c_{12} & \dots & c_{1N_h} & d_{11} & d_{12} & \dots & d_{1N_h} \\ c_{21} & c_{22} & \dots & c_{2N_h} & d_{21} & d_{22} & \dots & d_{2N_h} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ c_{N_h 1} & c_{N_h 2} & \dots & c_{N_h N_h} & d_{N_h 1} & d_{N_h 2} & \dots & d_{N_h N_h} \end{pmatrix} \begin{pmatrix} U_1(t) \\ U_2(t) \\ \vdots \\ U_{N_h}(t) \\ V_1(t) \\ V_2(t) \\ \vdots \\ V_{N_h}(t) \end{pmatrix} = \begin{pmatrix} F_1^1(t) \\ F_2^1(t) \\ \vdots \\ F_{N_h}^1(t) \\ F_1^2(t) \\ F_2^2(t) \\ \vdots \\ F_{N_h}^2(t) \end{pmatrix}, \quad (3.2.117)$$

where  $a_{ij}$ ,  $b_{ij}$ ,  $c_{ij}$  and  $d_{ij}$  are functions of time  $t$ . The vectors  $(U_1(t), U_2(t), \dots, U_{N_h}(t))^T$  and  $(V_1(t), V_2(t), \dots, V_{N_h}(t))^T$  are the solution vectors. We let

$$\begin{aligned} \mathbf{A} &= \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1N_h} \\ a_{21} & a_{22} & \dots & a_{2N_h} \\ \vdots & \vdots & \ddots & \vdots \\ a_{N_h 1} & a_{N_h 2} & \dots & a_{N_h N_h} \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} b_{11} & b_{12} & \dots & b_{1N_h} \\ b_{21} & b_{22} & \dots & b_{2N_h} \\ \vdots & \vdots & \ddots & \vdots \\ b_{N_h 1} & b_{N_h 2} & \dots & b_{N_h N_h} \end{pmatrix}, \\ \mathbf{C} &= \begin{pmatrix} c_{11} & c_{12} & \dots & c_{1N_h} \\ c_{21} & c_{22} & \dots & c_{2N_h} \\ \vdots & \vdots & \ddots & \vdots \\ c_{N_h 1} & c_{N_h 2} & \dots & c_{N_h N_h} \end{pmatrix}, \quad \mathbf{D} = \begin{pmatrix} d_{11} & d_{12} & \dots & d_{1N_h} \\ d_{21} & d_{22} & \dots & d_{2N_h} \\ \vdots & \vdots & \ddots & \vdots \\ d_{N_h 1} & d_{N_h 2} & \dots & d_{N_h N_h} \end{pmatrix}, \\ &\begin{cases} \mathbf{U}(t) = (U_1(t), U_2(t), \dots, U_{N_h}(t))^T, \\ \mathbf{V}(t) = (V_1(t), V_2(t), \dots, V_{N_h}(t))^T, \\ \mathbf{F}^1(t) = (F_1^1(t), F_2^1(t), \dots, F_{N_h}^1(t))^T, \\ \mathbf{F}^2(t) = (F_1^2(t), F_2^2(t), \dots, F_{N_h}^2(t))^T, \end{cases} \end{aligned}$$

and write (3.2.117) in block-vector form as

$$\begin{pmatrix} \mathbf{A}(t) & \mathbf{B}(t) \\ \mathbf{C}(t) & \mathbf{D}(t) \end{pmatrix} \begin{pmatrix} \mathbf{U}(t) \\ \mathbf{V}(t) \end{pmatrix} = \begin{pmatrix} \mathbf{F}^1(t) \\ \mathbf{F}^2(t) \end{pmatrix}. \quad (3.2.118)$$

We define the matrices and vectors by

$$\mathbf{A}(t) = \{a_{ij}(t)\} : a_{ij}(t) = \int_{\Omega_{h,t}} \left( 2 \frac{\partial \phi_i}{\partial x} \frac{\partial \phi_j}{\partial x} + \frac{\partial \phi_i}{\partial y} \frac{\partial \phi_j}{\partial y} \right) d\Omega_{h,t},$$

$$\mathbf{B}(t) = \{b_{ij}(t)\} : b_{ij}(t) = \int_{\Omega_{h,t}} \frac{\partial \phi_i}{\partial y} \frac{\partial \phi_j}{\partial x} d\Omega_{h,t},$$

$$\mathbf{C}(t) = \{c_{ij}(t)\} : c_{ij}(t) = \int_{\Omega_{h,t}} \frac{\partial \phi_i}{\partial x} \frac{\partial \phi_j}{\partial y} d\Omega_{h,t},$$

$$\mathbf{D}(t) = \{d_{ij}(t)\} : d_{ij}(t) = \int_{\Omega_{h,t}} \left( \frac{\partial \phi_i}{\partial x} \frac{\partial \phi_j}{\partial x} + 2 \frac{\partial \phi_i}{\partial y} \frac{\partial \phi_j}{\partial y} \right) d\Omega_{h,t}.$$

$$\mathbf{F}^1(t) = \{F_i^1(t)\} : F_i^1(t) = - \int_{\Omega_{h,t}} \left( f^h(\mathbf{x}(t)) \frac{\partial \phi_i}{\partial x} \right) d\Omega_{h,t} + \int_{\partial\Omega_{h,t}} (n_1 f^h(\mathbf{x}(t)) \phi_i) d\partial\Omega_{h,t},$$

$$\mathbf{F}^2(t) = \{F_i^2(t)\} : F_i^2(t) = - \int_{\Omega_{h,t}} \left( f^h(\mathbf{x}(t)) \frac{\partial \phi_i}{\partial y} \right) d\Omega_{h,t} + \int_{\partial\Omega_{h,t}} (n_2 f^h(\mathbf{x}(t)) \phi_i) d\partial\Omega_{h,t},$$

with

$$f^h = \delta(l) \eta_2 \sum_{j=1}^{N_h} \rho_j(t) \phi_j - \eta_1 \sum_{j=1}^{N_h} \omega_j(t) \phi_j,$$

We also note from (3.2.115) and (3.2.116) that  $\mathbf{C}(t) = (\mathbf{B}(t))^T$  and write (3.2.118) as

$$\begin{pmatrix} \mathbf{A}(t) & \mathbf{B}(t) \\ (\mathbf{B}(t))^T & \mathbf{D}(t) \end{pmatrix} \begin{pmatrix} \mathbf{U}(t) \\ \mathbf{V}(t) \end{pmatrix} = \begin{pmatrix} \mathbf{F}^1(t) \\ \mathbf{F}^2(t) \end{pmatrix}. \quad (3.2.119)$$

We let

$$\mathcal{A}(t) = \begin{pmatrix} \mathbf{A}(t) & \mathbf{B}(t) \\ (\mathbf{B}(t))^T & \mathbf{D}(t) \end{pmatrix}, \quad \boldsymbol{\xi}(t) = \begin{pmatrix} \mathbf{U}(t) \\ \mathbf{V}(t) \end{pmatrix}, \quad \mathbf{F} = \begin{pmatrix} \mathbf{F}^1(t) \\ \mathbf{F}^2(t) \end{pmatrix},$$

and have the following semi-discrete equation for the force balance equation

$$\mathcal{A}(t) \boldsymbol{\xi}(t) = \mathbf{F}(t). \quad (3.2.120)$$

### Semi-discrete equations for the coupled problem

The semi discrete equations for the reaction-advection-diffusion and force balance equations are of the form

$$\frac{d}{dt} (\mathbf{M}(t) \boldsymbol{\omega}(t)) = -\mathbf{S}(\rho(t)) \boldsymbol{\omega}(t), \quad (3.2.121)$$

and

$$\frac{d}{dt}(\mathbf{M}(t)\boldsymbol{\rho}(t)) = -(d\mathbf{K}(t) + e\mathbf{M}(t))\boldsymbol{\rho}(t) + a_{mon}(t)k_4\mathbf{H}(t) + k_3a_{mon}(t)\mathbf{L}(\rho(t)), \quad (3.2.122)$$

and

$$\mathcal{A}(t)\boldsymbol{\xi}(t) = \mathbf{F}(t). \quad (3.2.123)$$

### Fully discrete model

To obtain fully discrete equations for the coupled problem, we discretise the time interval  $(0, T]$  into a finite number  $N$  of uniform sub-intervals with sub-interval size  $\tau = t^{n+1} - t^n$  and write  $t^n = \tau n$ . From the semi-discrete equations

$$\frac{d}{dt}(\mathbf{M}(t)\boldsymbol{\omega}(t)) = -\mathbf{S}(\rho(t))\boldsymbol{\omega}(t),$$

and

$$\frac{d}{dt}(\mathbf{M}(t)\boldsymbol{\rho}(t)) = -(d\mathbf{K}(t) + e\mathbf{M}(t))\boldsymbol{\rho}(t) + a_{mon}(t)k_4\mathbf{H}(t) + k_3a_{mon}(t)\mathbf{L}(\rho(t)),$$

we employ the 2-SBDF as follows

$$\frac{3\mathbf{M}(t^{n+1})\boldsymbol{\omega}(t^{n+1}) - 4\mathbf{M}(t^n)\boldsymbol{\omega}(t^n) + \mathbf{M}(t^{n-1})\boldsymbol{\omega}(t^{n-1})}{2\tau} = -\mathbf{S}(\rho(t^n))\boldsymbol{\omega}(t^{n+1}),$$

and

$$\begin{aligned} \frac{3\mathbf{M}(t^{n+1})\boldsymbol{\rho}(t^{n+1}) - 4\mathbf{M}(t^n)\boldsymbol{\rho}(t^n) + \mathbf{M}(t^{n-1})\boldsymbol{\rho}(t^{n-1})}{2\tau} &= -d\mathbf{K}(t^{n+1})\boldsymbol{\rho}(t^{n+1}) \\ &+ 2(-e\mathbf{M}(t^n)\boldsymbol{\rho}(t^n) + k_4a_{mon}(t^n)\mathbf{H}(t^n) + k_3a_{mon}(t^n)\mathbf{L}(\rho(t^n))) \\ &- (-e\mathbf{M}(t^{n-1})\boldsymbol{\rho}(t^{n-1}) + k_4a_{mon}(t^{n-1})\mathbf{H}(t^{n-1}) + k_3a_{mon}(t^{n-1})\mathbf{L}(\rho(t^{n-1}))) . \end{aligned}$$

For (3.2.123), we compute  $\mathbf{A}(t)$  and  $\mathbf{F}(t)$  at time step  $t^n$  as follows

$$\mathcal{A}(t^n)\boldsymbol{\xi}^{n+1} = \mathbf{F}(t^n).$$

The above equations yield the following fully discrete equations

$$(3\mathbf{M}(t^{n+1}) + 2\tau\mathbf{S}(\rho(t^n)))\boldsymbol{\omega}(t^{n+1}) = 4\mathbf{M}(t^n)\boldsymbol{\omega}(t^n) - \mathbf{M}(t^{n-1})\boldsymbol{\omega}(t^{n-1}), \quad (3.2.124)$$

$$\begin{aligned} (3\mathbf{M}(t^{n+1}) + 2d\tau\mathbf{K}(t^{n+1}))\boldsymbol{\rho}(t^{n+1}) &= 4\mathbf{M}(t^n)\boldsymbol{\rho}(t^n) - \mathbf{M}(t^{n-1})\boldsymbol{\rho}(t^{n-1}) \\ &+ 4\tau(-e\mathbf{M}(t^n)\boldsymbol{\rho}(t^n) + k_4a_{mon}(t^n)\mathbf{H}(t^n) + k_3a_{mon}(t^n)\mathbf{L}(\rho(t^n))) \\ &- 2\tau(-e\mathbf{M}(t^{n-1})\boldsymbol{\rho}(t^{n-1}) + k_4a_{mon}(t^{n-1})\mathbf{H}(t^{n-1}) + k_3a_{mon}(t^{n-1})\mathbf{L}(\rho(t^{n-1}))) , \end{aligned} \quad (3.2.125)$$

$$\mathcal{A}(t^n)\boldsymbol{\xi}^{n+1} = \mathbf{F}(t^n), \quad (3.2.126)$$

with  $\tau$  as the time-step size,  $\boldsymbol{\rho}(t^{n+1})$  and  $\boldsymbol{\omega}(t^{n+1})$  are the actin filaments and myosin II solutions at time  $t^{n+1}$  and  $\boldsymbol{\xi}^{n+1}$  is the velocity solutions at time  $t^{n+1}$ .

### One-step backward Euler scheme

We will need solutions at the last two time steps in order to implement the 2-SBDF. We therefore implement a one-step backward Euler method to solve for  $\boldsymbol{\rho}(t^1)$  and  $\boldsymbol{\omega}(t^1)$  and then proceed with 2-SBDF scheme. A one-step backward Euler scheme for (3.2.121) and (3.2.122) is given by

$$(\mathbf{M}(t^1) + \tau \mathbf{S}(\rho(t^0))) \boldsymbol{\omega}(t^1) = \mathbf{M}(t^0) \boldsymbol{\omega}(t^0), \quad (3.2.127)$$

$$(\mathbf{M}(t^1) + d\tau \mathbf{K}(t^1)) \boldsymbol{\rho}(t^1) = (1 - e\tau) \mathbf{M}(t^0) \boldsymbol{\rho}(t^0) + \tau k_4 a_{mon}(t^0) \mathbf{H}(t^0) + \tau k_3 a_{mon}(t^0) \mathbf{L}(\rho(t^0)). \quad (3.2.128)$$

### Displacements of the nodes

We now describe how the nodes of the mesh will be displaced to obtain a new nodal location. The position of any new node will be a function of its current position and the amount of displacement it has achieved. Let  $t^{n+1} = t^n + \tau$  and consider the points  $\mathbf{x}(t^n) \in \boldsymbol{\Omega}_{h,t^n}$  and  $\mathbf{x}(t^{n+1}) \in \boldsymbol{\Omega}_{h,t^{n+1}}$  in the respective domains. We can define a first order linear approximation of the flow velocity as follows:

$$\boldsymbol{\xi}(\mathbf{x}(t^n), t^n) = \frac{\mathbf{x}(t^{n+1}) - \mathbf{x}(t^n)}{\tau}. \quad (3.2.129)$$

This means that the new domain can be approximated by

$$\mathbf{x}(t^{n+1}) = \mathbf{x}(t^n) + \tau \boldsymbol{\xi}(\mathbf{x}(t^n)), \quad (3.2.130)$$

where  $\boldsymbol{\xi}(\mathbf{x}(t^n))$  is the solution of the force balance equation at time  $t^n$ . We note that  $\tau \boldsymbol{\xi}(\mathbf{x}(t^n))$  is the displacement from point  $\mathbf{x}(t^n)$  to  $\mathbf{x}(t^{n+1})$ . Thus, the new nodal position will be given as the sum of the current node and its displacement.

### Quadrature formula

It now remains to discretise the integrals in the matrices and vectors. To implement the discretisation, we use **deal.II** library [Bangerth et al. \(2007\)](#). Let each quadrilateral be mapped onto the canonical quadrilateral  $\hat{K}$  by a bi-linear transformation. We split the integrals over the entire domain into integrals over each cell  $K_i$  and map each cell to  $\hat{K}$ . The global matrices and vectors will therefore be given as the contribution over each cell. The quadrature formula for the matrices and vectors are given by

$$\begin{aligned} m_{ij}(t) &= \sum_{K \in \boldsymbol{\Omega}_{h,t}} \int_K \phi_i(\mathbf{x}(t), t) \cdot \phi_j(\mathbf{x}(t), t) dK \\ &= \sum_{K \in \boldsymbol{\Omega}_{h,t}} \int_{\hat{K}} \hat{\phi}_i(\hat{\mathbf{x}}(t), t) \cdot \hat{\phi}_j(\hat{\mathbf{x}}(t), t) |\mathbf{J}_K(\hat{\mathbf{x}}(t))| d\hat{K} \\ &= \sum_{K \in \boldsymbol{\Omega}_{h,t}} \sum_q \hat{\phi}_i(\hat{\mathbf{x}}_q(t), t) \cdot \hat{\phi}_j(\hat{\mathbf{x}}_q(t), t) |\mathbf{J}_K(\hat{\mathbf{x}}_q(t))| \mathbf{w}(\hat{\mathbf{x}}_q(t)), \end{aligned}$$

$$\begin{aligned}
k_{ij}(t) &= \sum_{K \in \Omega_{h,t}} \int_K \nabla \phi_i(\mathbf{x}(t), t) \cdot \nabla \phi_j(\mathbf{x}(t), t) dK \\
&= \sum_{K \in \Omega_{h,t}} \int_{\hat{K}} \hat{\nabla} \hat{\phi}_i(\hat{\mathbf{x}}(t), t) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}(t), t) \cdot \hat{\nabla} \hat{\phi}_j(\hat{\mathbf{x}}(t), t) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}(t), t) |\mathbf{J}_K(\hat{\mathbf{x}}(t))| d\hat{K} \\
&= \sum_{K \in \Omega_{h,t}} \sum_q \hat{\nabla} \hat{\phi}_i(\hat{\mathbf{x}}_q(t), t) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}_q(t), t) \cdot \hat{\nabla} \hat{\phi}_j(\hat{\mathbf{x}}_q(t), t) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}_q(t), t) |\mathbf{J}_K(\hat{\mathbf{x}}_q(t))| \mathbf{w}(\hat{\mathbf{x}}_q(t)),
\end{aligned}$$

$$\begin{aligned}
s_{ij}(t) &= \sum_{K \in \Omega_{h,t}} \int_K \frac{\nabla \phi_i(\mathbf{x}(t), t) \cdot \nabla \phi_j(\mathbf{x}(t), t)}{1 + a \sum_{k=1}^{N_h} \rho_k(t) \phi_k(\mathbf{x})} dK \\
&= \sum_{K \in \Omega_{h,t}} \int_{\hat{K}} \frac{\hat{\nabla} \hat{\phi}_i(\hat{\mathbf{x}}(t), t) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}(t), t) \cdot \hat{\nabla} \hat{\phi}_j(\hat{\mathbf{x}}(t), t) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}(t), t)}{1 + a \sum_{k=1}^{N_h} \rho_k(t) \hat{\phi}_k(\hat{\mathbf{x}}(t), t)} |\mathbf{J}_K(\hat{\mathbf{x}}(t))| d\hat{K} \\
&= \sum_{K \in \Omega_{h,t}} \sum_q \frac{\hat{\nabla} \hat{\phi}_i(\hat{\mathbf{x}}_q(t), t) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}_q(t), t) \cdot \hat{\nabla} \hat{\phi}_j(\hat{\mathbf{x}}_q(t), t) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}_q(t), t)}{1 + a \sum_{k=1}^{N_h} \rho_k(t) \hat{\phi}_k(\hat{\mathbf{x}}_q(t))} |\mathbf{J}_K(\hat{\mathbf{x}}_q(t))| \mathbf{w}(\hat{\mathbf{x}}_q(t)),
\end{aligned}$$

$$\begin{aligned}
a_{ij}(t) &= \sum_{K \in \Omega_{h,t}} \int_K \left( 2 \frac{\partial \phi_i}{\partial x} \frac{\partial \phi_j}{\partial x} + \frac{\partial \phi_i}{\partial y} \frac{\partial \phi_j}{\partial y} \right) d\Omega_{h,t} \\
&= \sum_{K \in \Omega_{h,t}} \int_{\hat{K}} \hat{\nabla} \hat{\phi}_i(\hat{\mathbf{x}}(t), t) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}(t)) \mathbf{G}_1 \cdot \hat{\nabla} \hat{\phi}_j(\hat{\mathbf{x}}(t), t) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}(t)) |\mathbf{J}_K(\hat{\mathbf{x}}(t))| d\hat{K} \\
&= \sum_{K \in \Omega_{h,t}} \sum_q \hat{\nabla} \hat{\phi}_i(\hat{\mathbf{x}}_q(t), t) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}_q(t)) \mathbf{G}_1 \cdot \hat{\nabla} \hat{\phi}_j(\hat{\mathbf{x}}_q(t), t) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}_q(t)) |\mathbf{J}_K(\hat{\mathbf{x}}_q(t))| \mathbf{w}(\hat{\mathbf{x}}_q(t)),
\end{aligned}$$

$$\begin{aligned}
b_{ij}(t) &= \sum_{K \in \Omega_{h,t}} \int_K \frac{\partial \phi_i}{\partial y} \frac{\partial \phi_j}{\partial x} dK \\
&= \sum_{K \in \Omega_{h,t}} \int_{\hat{K}} \hat{\nabla} \hat{\phi}_i(\hat{\mathbf{x}}(t), t) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}(t)) \mathbf{G}_3 \cdot \hat{\nabla} \hat{\phi}_j(\hat{\mathbf{x}}(t), t) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}(t)) \mathbf{G}_4 |\mathbf{J}_K(\hat{\mathbf{x}}(t))| d\hat{K} \\
&= \sum_{K \in \Omega_{h,t}} \sum_q \hat{\nabla} \hat{\phi}_i(\hat{\mathbf{x}}_q(t), t) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}_q(t)) \mathbf{G}_3 \cdot \hat{\nabla} \hat{\phi}_j(\hat{\mathbf{x}}_q(t), t) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}_q(t)) \mathbf{G}_4 |\mathbf{J}_K(\hat{\mathbf{x}}_q(t))| \mathbf{w}(\hat{\mathbf{x}}_q(t)),
\end{aligned}$$

$$\begin{aligned}
d_{ij}(t) &= \sum_{K \in \Omega_{h,t}} \int_K \left( \frac{\partial \phi_i}{\partial x} \frac{\partial \phi_j}{\partial x} + 2 \frac{\partial \phi_i}{\partial y} \frac{\partial \phi_j}{\partial y} \right) dK \\
&= \sum_{K \in \Omega_{h,t}} \int_{\hat{K}} \hat{\nabla} \hat{\phi}_i(\hat{\mathbf{x}}(t), t) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}(t)) \cdot \hat{\nabla} \hat{\phi}_j(\hat{\mathbf{x}}(t), t) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}(t)) \mathbf{G}_2 |\mathbf{J}_K(\hat{\mathbf{x}}(t))| d\hat{K} \\
&= \sum_{K \in \Omega_{h,t}} \sum_q \hat{\nabla} \hat{\phi}_i(\hat{\mathbf{x}}_q(t), t) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}_q(t)) \cdot \hat{\nabla} \hat{\phi}_j(\hat{\mathbf{x}}_q(t), t) \mathbf{J}_K^{-1}(\hat{\mathbf{x}}_q(t)) \mathbf{G}_2 |\mathbf{J}_K(\hat{\mathbf{x}}_q(t))| \mathbf{w}(\hat{\mathbf{x}}_q(t)),
\end{aligned}$$

$$\begin{aligned}
h_i(t) &= \sum_{K \in \Omega_{h,t}} \int_K \phi_i(\mathbf{x}(t), t) dK = \sum_{K \in \Omega_{h,t}} \int_{\hat{K}} \hat{\phi}_i(\hat{\mathbf{x}}(t), t) |\mathbf{J}_K(\hat{\mathbf{x}}(t))| d\hat{K} \\
&= \sum_{K \in \Omega_{h,t}} \sum_q \hat{\phi}_i(\hat{\mathbf{x}}_q(t), t) |\mathbf{J}_K(\hat{\mathbf{x}}_q(t))| \mathbf{w}(\hat{\mathbf{x}}_q(t)),
\end{aligned}$$

$$\begin{aligned}
l_i(t) &= \sum_{K \in \Omega_{h,t}} \int_K \frac{\phi_i(\mathbf{x}(t), t) \left( \sum_{k=1}^{N_h} \rho_k(t) \phi_k(\mathbf{x}(t), t) \right)^2}{1 + \left( \sum_{k=1}^{N_h} \rho_k(t) \phi_k(\mathbf{x}(t), t) \right)^2} dK \\
&= \sum_{K \in \Omega_{h,t}} \int_{\hat{K}} \frac{\hat{\phi}_i(\hat{\mathbf{x}}(t), t) \left( \sum_{k=1}^{N_h} \rho_k(t) \hat{\phi}_k(\hat{\mathbf{x}}(t), t) \right)^2}{1 + \left( \sum_{k=1}^{N_h} \rho_k(t) \hat{\phi}_k(\hat{\mathbf{x}}(t), t) \right)^2} |\mathbf{J}_K(\hat{\mathbf{x}}(t))| d\hat{K} \\
&= \sum_{K \in \Omega_{h,t}} \sum_q \frac{\hat{\phi}_i(\hat{\mathbf{x}}_q(t), t) \left( \sum_{k=1}^{N_h} \rho_k(t) \hat{\phi}_k(\hat{\mathbf{x}}_q(t), t) \right)^2}{1 + \left( \sum_{k=1}^{N_h} \rho_k(t) \hat{\phi}_k(\hat{\mathbf{x}}_q(t), t) \right)^2} |\mathbf{J}_K(\hat{\mathbf{x}}_q(t))| \mathbf{w}(\hat{\mathbf{x}}_q(t)).
\end{aligned}$$

$$\begin{aligned}
F_i^1(t) &= - \sum_{K \in \Omega_{h,t}} \int_K \left( f^h(\mathbf{x}(t)) \frac{\partial \phi_i}{\partial x} \right) dK + \sum_{Z=K \cap \partial \Omega_{h,t}} \int_Z (n_1 f^h(\mathbf{x}(t)) \phi_i) dZ \\
&= - \sum_{K \in \Omega_{h,t}} \int_{\hat{K}} f^h(\hat{\mathbf{x}}(t)) \left( \sum_{l=1}^2 \frac{\partial \hat{\phi}_i(\hat{\mathbf{x}}(t))}{\partial \zeta_l} \frac{\partial \zeta_l}{\partial x} \right) |\mathbf{J}_K(\hat{\mathbf{x}}(t))| d\hat{K} \\
&\quad + \sum_{Z=K \cap \partial \Omega_{h,t}} \int_{\hat{K}} \left( n_1 f^h(\hat{\mathbf{x}}(t)) \hat{\phi}_i(\hat{\mathbf{x}}(t)) \right) |\mathbf{J}_Z(\hat{\mathbf{x}}(t))| d\hat{Z} \\
&= - \sum_{K \in \Omega_{h,t}} \sum_q f^h(\hat{\mathbf{x}}_q(t)) \left( \sum_{l=1}^2 \frac{\partial \hat{\phi}_i(\hat{\mathbf{x}}_q(t))}{\partial \zeta_l} \frac{\partial \zeta_l}{\partial x} \right) |\mathbf{J}_K(\hat{\mathbf{x}}_q(t))| \mathbf{w}(\hat{\mathbf{x}}_q(t)) \\
&\quad + \sum_{Z=K \cap \partial \Omega_{h,t}} \sum_q \left( n_1 f^h(\hat{\mathbf{x}}_q(t)) \hat{\phi}_i(\hat{\mathbf{x}}_q(t)) \right) |\mathbf{J}_Z(\hat{\mathbf{x}}_q(t))| \mathbf{w}(\hat{\mathbf{x}}_q(t)),
\end{aligned}$$

and

$$\begin{aligned}
F_i^2(t) &= - \sum_{K \in \Omega_{h,t}} \int_K \left( f^h(\mathbf{x}(t)) \frac{\partial \phi_i}{\partial y} \right) dK + \sum_{Z=K \cap \partial \Omega_{h,t}} \int_Z (n_2 f^h(\mathbf{x}(t)) \phi_i) dZ \\
&= - \sum_{K \in \Omega_{h,t}} \int_{\hat{K}} f^h(\hat{\mathbf{x}}(t)) \left( \sum_{l=1}^2 \frac{\partial \hat{\phi}_i(\hat{\mathbf{x}}(t))}{\partial \zeta_l} \frac{\partial \zeta_l}{\partial y} \right) |\mathbf{J}_K(\hat{\mathbf{x}}(t))| d\hat{K} \\
&\quad + \sum_{Z=K \cap \partial \Omega_{h,t}} \int_{\hat{K}} \left( n_2 f^h(\hat{\mathbf{x}}(t)) \hat{\phi}_i(\hat{\mathbf{x}}(t)) \right) |\mathbf{J}_Z(\hat{\mathbf{x}}(t))| d\hat{Z} \\
&= - \sum_{K \in \Omega_{h,t}} \sum_q f^h(\hat{\mathbf{x}}_q(t)) \left( \sum_{l=1}^2 \frac{\partial \hat{\phi}_i(\hat{\mathbf{x}}_q(t))}{\partial \zeta_l} \frac{\partial \zeta_l}{\partial y} \right) |\mathbf{J}_K(\hat{\mathbf{x}}_q(t))| \mathbf{w}(\hat{\mathbf{x}}_q(t)) \\
&\quad + \sum_{Z=K \cap \partial \Omega_{h,t}} \sum_q \left( n_2 f^h(\hat{\mathbf{x}}_q(t)) \hat{\phi}_i(\hat{\mathbf{x}}_q(t)) \right) |\mathbf{J}_Z(\hat{\mathbf{x}}_q(t))| \mathbf{w}(\hat{\mathbf{x}}_q(t)),
\end{aligned}$$

with

$$f^h = \delta(l) \eta_2 \sum_{j=1}^{N_h} \rho_j(t) \phi_j - \eta_1 \sum_{j=1}^{N_h} \omega_j(t) \phi_j,$$

$$\mathbf{G}_1 = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}, \quad \mathbf{G}_2 = \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix}, \quad \mathbf{G}_3 = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}, \quad \mathbf{G}_4 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}.$$

To implement the discretisation, we use **deal.II** library [Bangerth et al. \(2007\)](#). We note that both the mass and stiffness matrices  $\mathbf{M}(t)$  and  $\mathbf{K}(t)$  respectively are large, sparse, symmetric and positive definite while matrices  $\mathbf{A}(t)$ ,  $\mathbf{B}(t)$ ,  $\mathbf{D}(t)$  and  $\mathbf{S}(t)$  are large, sparse and non-symmetric. A sparse matrix has very few nonzero entries. Iterative methods are the best suited for solving such systems [Saad \(2003\)](#). Here, we will use iterative methods that use preconditioners. The rate of convergence of an iterative method depends on the spectrum of the matrices, that is, the rate of convergence depends on the condition number of the matrices [Saad \(2003\)](#). A preconditioner gives rise to a more favourable spectrum of the resultant matrices [Barrett et al. \(1994\)](#); [Madzvamuse \(2000\)](#). Since the matrices  $\mathbf{M}(t)$  and  $\mathbf{K}(t)$  are both symmetric and positive definite, the system (3.2.125) can be solved using a preconditioned conjugate gradient method (PCG) with a diagonal preconditioner [Saad \(2003\)](#); [Barrett et al. \(1994\)](#). The systems (3.2.124) and (3.2.126) will be solved using a generalised minimal residual method (GMRES) with a diagonal preconditioner as illustrated in [Freund et al. \(1992\)](#). Since the computational domain will change at each time  $t$ , the matrices and vectors have to be updated at each time  $t$ . We summarise the numerical algorithm that we use as follows:

1. Initialise parameters to be used and the initial conditions for the variables.
2. Discretise the initial domain  $\Omega_{h,t^0}$ .
3. Assemble matrices at initial time  $t^0$ .
4. Solve (3.2.127) for  $\omega^1$ , (3.2.128) for  $\rho^1$  and one step of (3.2.126) for  $\xi^1$ .
5. **WHILE**  $t < \text{Final time}$ 
  - (a) Compute the computational domain  $\Omega_{h,t^n}$ .
  - (b) Assemble matrices and vectors at time  $t^n$ .
  - (c) Solve the discrete system (3.2.126) for the force balance equation to obtain solution at  $t^{n+1}$ .
  - (d) Compute the new domain  $\Omega_{h,t^{n+1}}$ .
  - (e) Assemble new matrices and vectors at time  $t^{n+1}$ .
  - (f) Solve the discrete systems (3.2.124) and (3.2.125) for the reaction-advection-diffusion equations to obtain solution at  $t^{n+1}$ .
  - (g)  $t = t + \tau$ .
  - (h) Old solution=new solution.
6. **END**.

### Computation of the unit normal vector to the boundary

In the integration over the boundary of the domain, we need to compute the unit normals  $\mathbf{n}(\mathbf{x}, t) = (n_1(\mathbf{x}, t), n_2(\mathbf{x}, t))$  to the boundary. We hereby describe techniques for computing the normals at

a point on the boundary  $\partial\Omega_{h,t}$ . We check whether the numerically computed normals converge to the analytical normals as the number of boundary nodes increase, that is, as  $h$  tends to zero.

On the boundary  $\partial\Omega_{h,t}$  of the domain  $\Omega_{h,t}$ , any two nodes are connected by a straight line. For a general point  $P_i$  ( $i = 1, 2, \dots, n_d$ ) where  $n_d$  is the total number of nodes on the boundary  $\partial\Omega_{h,0}$ , we have the outward pointing unit normal vector to the line joining any two points as

$$\mathbf{n} = \left( \frac{y_i - y_{i+1}}{\sqrt{(y_i - y_{i+1})^2 + (x_{i+1} - x_i)^2}}, \frac{x_{i+1} - x_i}{\sqrt{(y_i - y_{i+1})^2 + (x_{i+1} - x_i)^2}} \right), \quad (3.2.131)$$

Zill et al. (2011). We note that there will be two outward pointing unit normals at each node  $P_i$  on the boundary as a result of intersection of the two line segments. The numerically computed unit normal will therefore be obtained by using the parallelogram law of vector addition. The equation of the unit circle centred at the origin is given by  $f(x, y) = x^2 + y^2 - 1$ . The exact outward pointing unit normal at any point  $(x, y)$  of the unit circle is given by  $\mathbf{n}(x, y) = \frac{\nabla f(x, y)}{|\nabla f(x, y)|}$  Zill et al. (2011). To check the accuracy of the computed normals  $\mathbf{n}_{num}$  at the nodal points, we compute the  $L^2$  norm  $\|\frac{\mathbf{n}_{num} - \mathbf{n}_{exact}}{\tau}\|$  where  $\mathbf{n}_{exact}$  is the exact normal at the nodes. Table 3.4 shows that as the number of refinements increase, the numerically computed normal converge to the exact normal to the unit circle.

Mesh	mesh diameter $h$	$\ \frac{\mathbf{n}_{num} - \mathbf{n}_{exact}}{\tau}\ _{L^2}$
1	1.08239	0.517638
2	0.709704	0.234199
3	0.404102	0.114279
4	0.211681	0.056795
5	0.109097	0.028355
6	0.055126	0.014172

Table 3.4: The numerical normals to a circle converge to the exact normal to a circle as the mesh is refined.

### Numerical simulation of cell movement

Here, we present some numerical results for the non-dimensionalised viscous model. The solution for this model is in form of F-actin and myosin II concentration and the speed of the cell. F-actin and myosin II solution are the solution for the reaction-advection-diffusion equations while speeds of the cell come from solution of the force balance equation. We begin simulations on a unit disk to represent the cell at initial time with zero initial speed. We consider different data for the initial conditions of F-actin and myosin II concentrations. The first set of initial conditions for the concentrations are: for myosin II, we consider random perturbations about  $\rho_m = 1.0$  as is in Figure 3.14 and for F-actin, a nonzero concentrations of  $\rho_a = 1.0$  only in one half of the cell as shown in Figure 3.15 while the second set of initial conditions are random perturbations about  $\rho_a = \rho_m = 1.0$  as is in Figures 3.17, 3.18, 3.20, 3.21 and 3.22. We also vary the following



parameters at a time while keeping all the other parameters constant: (i) total amount of actin  $\rho_a^{tot}$  Figures 3.20, 3.21, 3.22 and 3.23 and (ii) contraction coefficient for myosin II  $\eta_m$  Figure 3.23.

Actin polymerisation causes polymerisation stress at the cell periphery which results in expansion of the cell. We observe that the initial conditions chosen determine the dynamics of F-actin and cell shape. Choosing a perturbation about  $\rho_a = 1.0$  as the initial condition for  $\rho_a$  variable leads to a uniform expansion of the cell as shown in Figures 3.19 and 3.22. For this case, actin polymerises around the cell periphery and depolymerises in most of the other parts of the cell. Choosing a non-zero initial concentrations of F-actin only in one half of the cell leads to symmetry breaking where the cell identifies its front and rear. Actin polymerises at the cell front and results in protrusion stress at the front while it depolymerises at the rest of the cell. This leads to an irregular expansion of the cell as shown in Figure 3.16 and hence in a directed migration of the cell towards the direction with high F-actin concentrations.

We note that in our model, myosin II only diffuses inside the cell and exerts contractile stress in the cell. Its total concentration in the entire cell is conserved. Myosin II exerts contractile stress in regions where protrusive stress does not act. This stress helps in propelling the cell forward.

Actin changes from the active state (F-actin) to inactive state (G-actin) and vice-versa through polymerisation and depolymerisation processes and hence the total amount of actin is conserved at all time. F-actin assembles together causing expansive stress on the cell. We varied the total amount of actin in the cell and observed that the more the total amount of actin, the more the expansion of the cell. Figures 3.16, 3.19, 3.22 and 3.23 show change in the area of the motile cell with time. Figures 3.14, 3.15 and 3.16 show one set of solution with initial data for myosin II as random perturbation about  $\rho_m = 1$ , initial condition for  $\rho_a$  nonzero only in one half of the cell and all other parameters as in Table 3.3 while Figures 3.17, 3.18 and 3.19 is a solution set with initial data for the concentrations as random perturbation about  $\rho_m = 1$  and  $\rho_a = 1$  while all other parameters as in Table 3.3. Figures 3.20, 3.21 and 3.22 is a solution set with initial data for the concentrations as random perturbation about  $\rho_m = 1$  and  $\rho_a = 1$ , total amount of actin increased to  $\rho_a^{tot} = 16$  while all other parameters unchanged, Figure 3.23 part (a) is a graph when contraction coefficient is chosen to be  $\eta_1 = 0.2$  while all other parameters unchanged and Figure 3.23 part (b) is the graph of area of cell with time when total amount of actin is reduced to  $\rho_a^{tot} = 5$ .

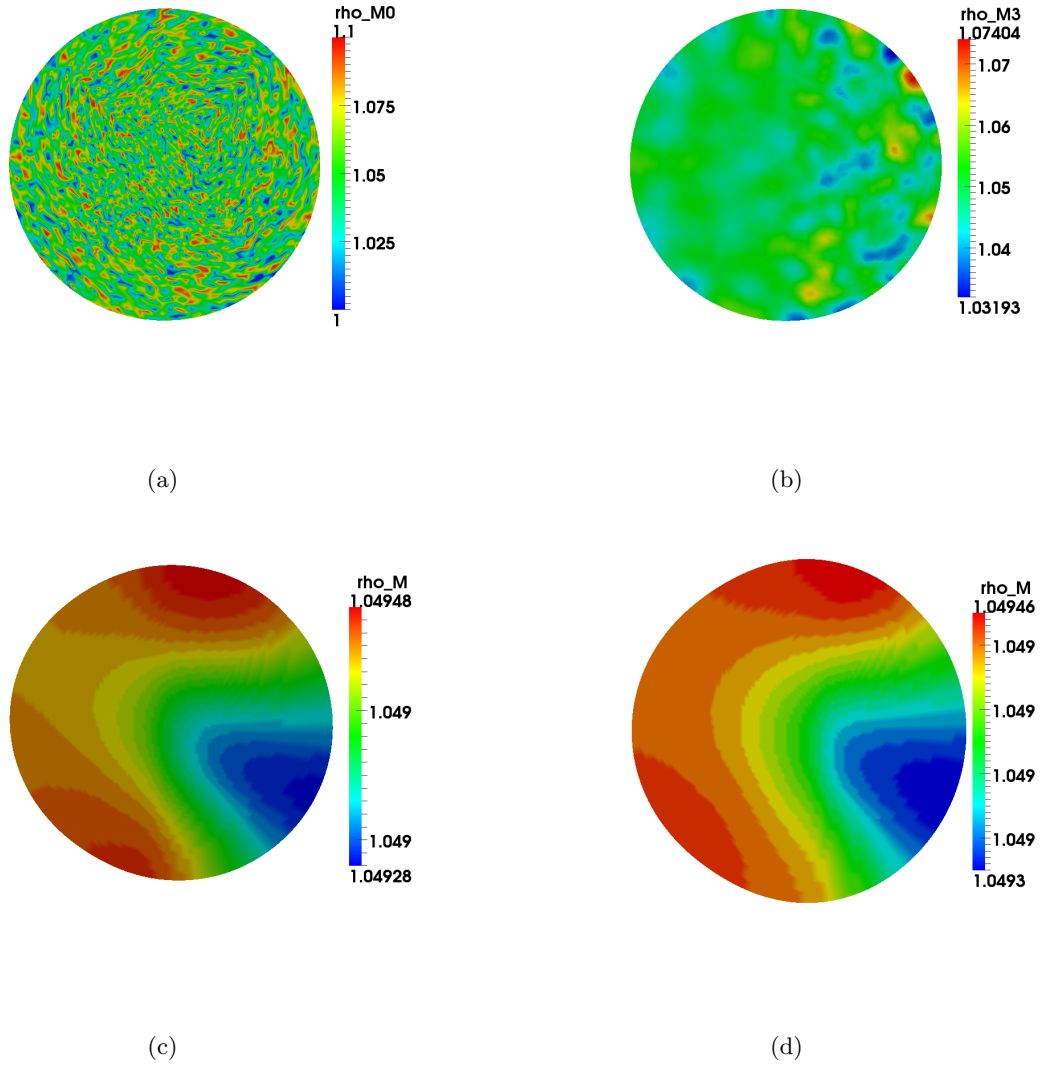


Figure 3.14: Graphical displays of the numerical results of the myosin II concentration  $\rho_m$  using a 2-SBDF scheme with  $\tau = 0.001$ . Blue signifies lowest values while red highest values. (a) Initial condition as random perturbation about  $\rho_m = 1.0$  (b)  $\rho_m$  at time  $t = 0.004$ , (c)  $\rho_m$  at time  $t = 2$  and (d) solution at final time  $t = 4$ .

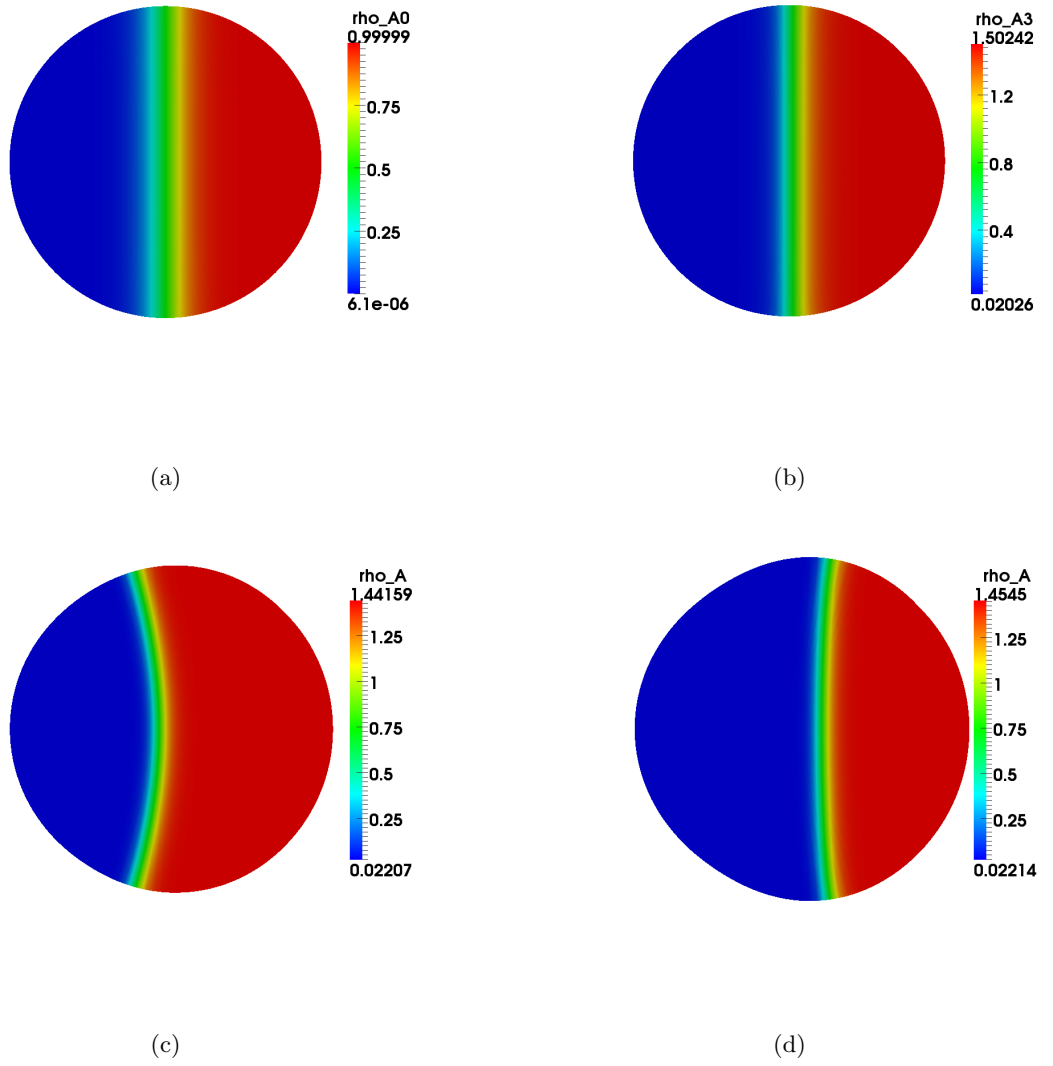


Figure 3.15: Graphical displays of the numerical results of the F-actin concentration  $\rho_a$  using a 2-SBDF scheme with  $\tau = 0.001$ . Blue signifies lowest values while red highest values. (a) Initial condition as non-zero only in one half of the cell (b)  $\rho_a$  at time  $t = 0.004$ , (c)  $\rho_a$  at time  $t = 2$  and (d) solution at final time  $t = 4$ .

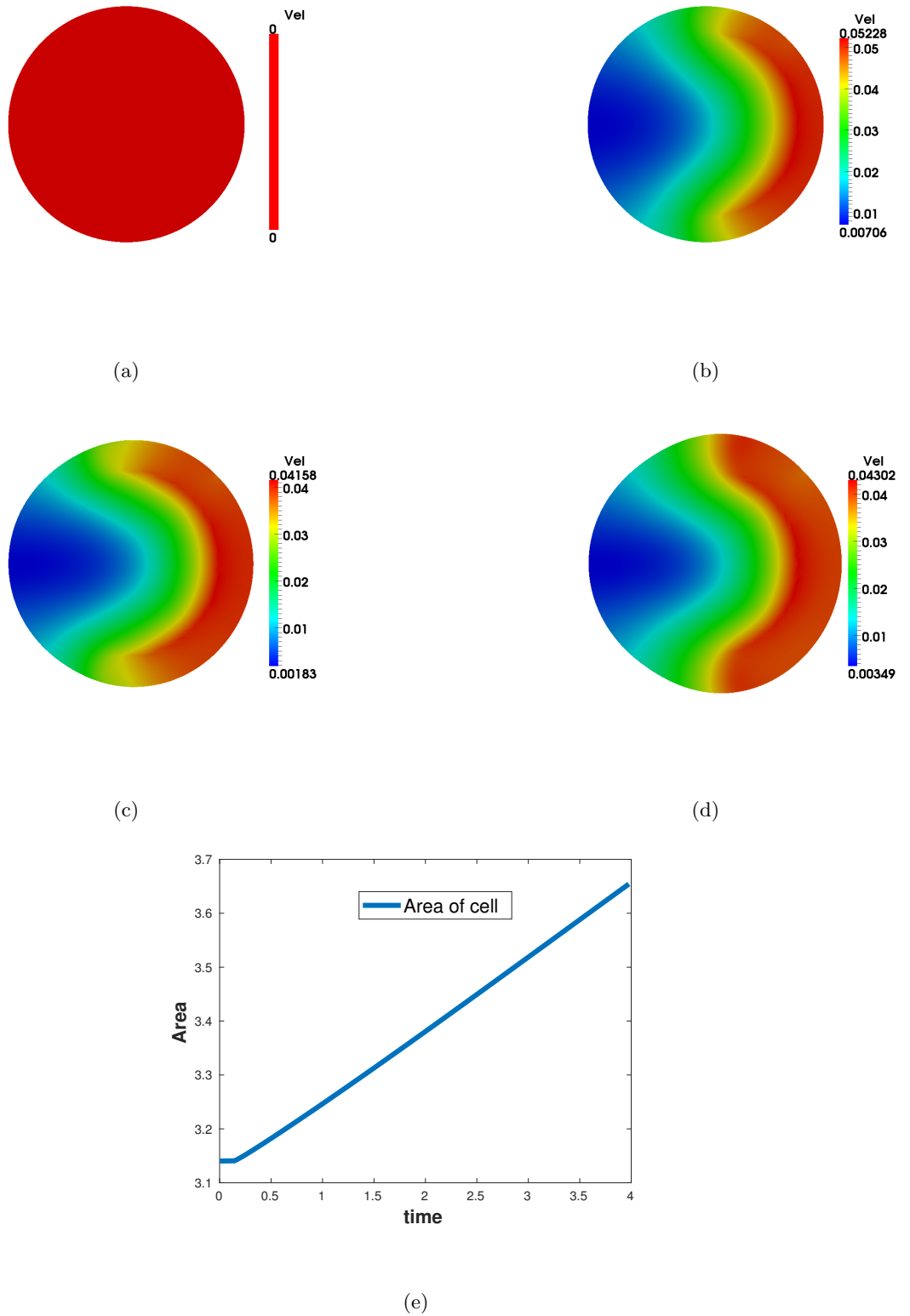


Figure 3.16: Graphical display of the speed of the cell as the solution of the force balance equation for the cell. Blue signifies lowest values while red highest values. (a) Cell at initial stationary state (b) cell speed at time  $t = 0.004$ , (c) cell speed at time  $t = 2$ , (d) cell speed at final time  $t = 4$  and (e) area of the evolving cell as a function of time showing increase in the area.

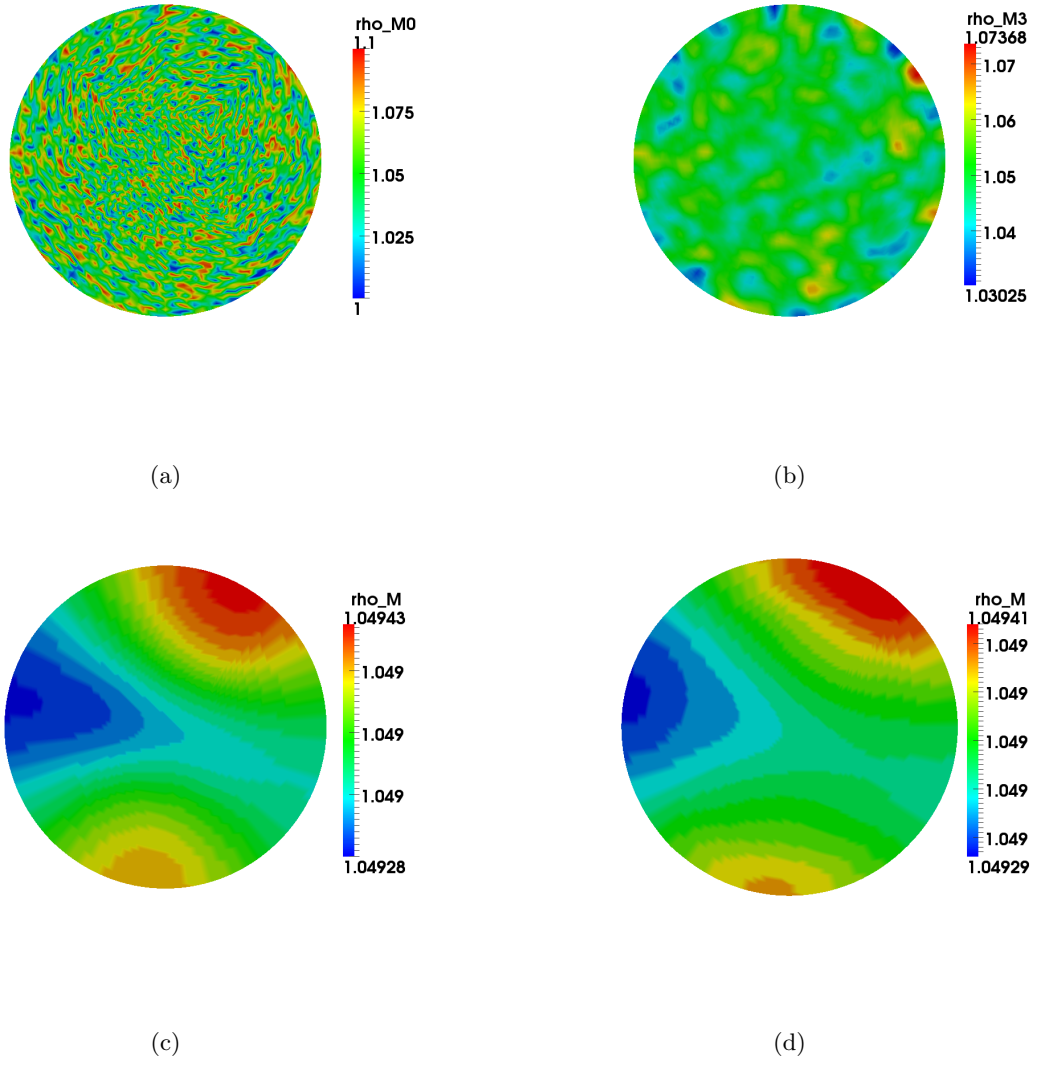


Figure 3.17: Solutions for the  $\rho_m$  variable using a 2-SBDF scheme with  $\tau = 0.001$ . Blue signifies lowest values while red highest values. (a) Initial condition as random perturbation about  $\rho_m = 1.0$ , (b)  $\rho_m$  at time  $t = 0.004$ , (c)  $\rho_m$  at time  $t = 2$  and (d) solution at final time  $t = 4$ .

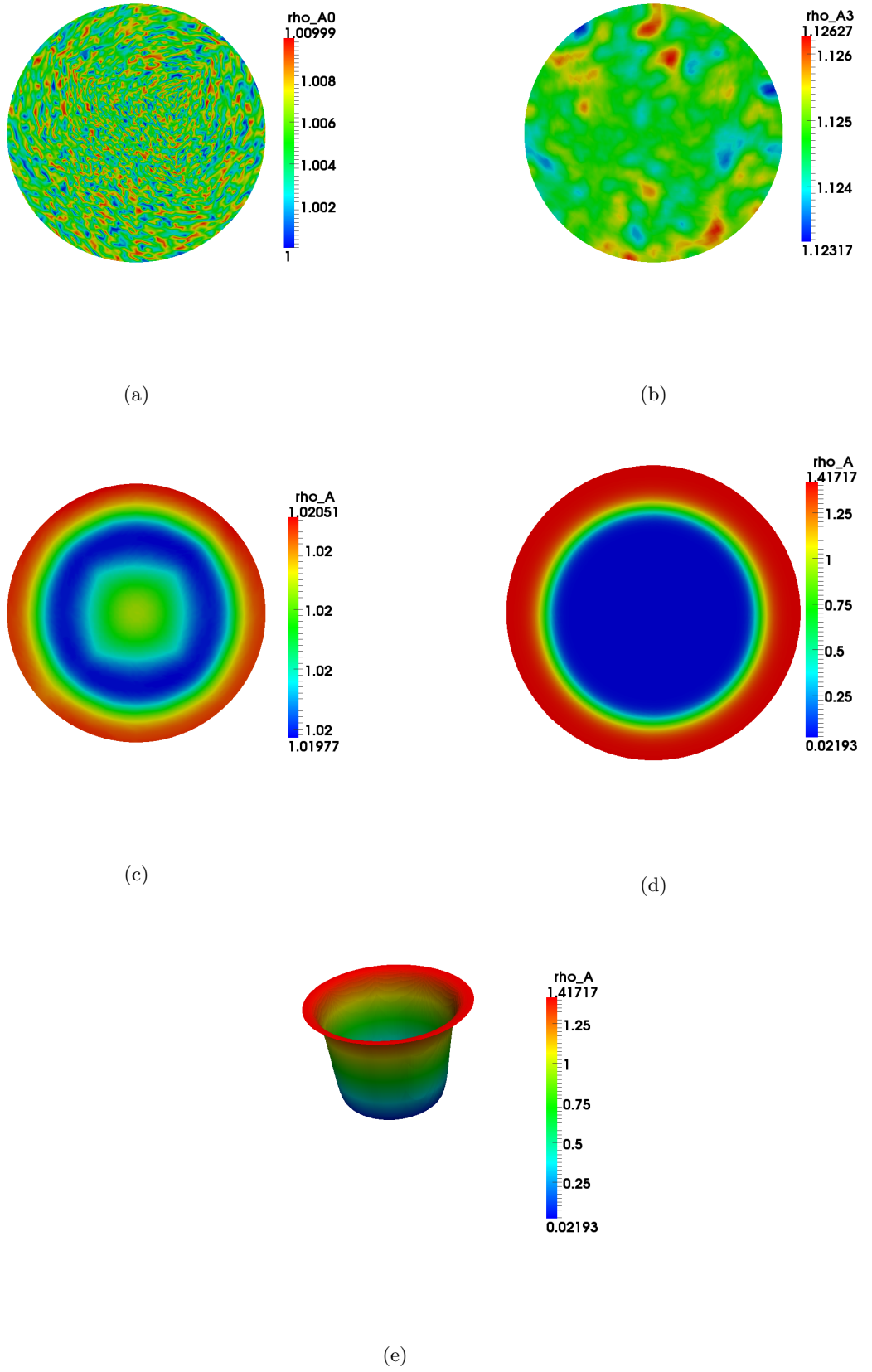


Figure 3.18: Solutions for the  $\rho_a$  variable using 2-SBDF scheme with  $\tau = 0.001$ . Blue signifies lowest values while red highest values. (a) Initial condition as random perturbation about  $\rho_a = 1.0$  (b)  $\rho_a$  at time  $t = 0.004$ , (c)  $\rho_a$  at time  $t = 2$  (d)  $\rho_a$  solution at final time  $t = 4$  and (e) surface plot for the  $\rho_a$  solution at final time  $t = 4$ .

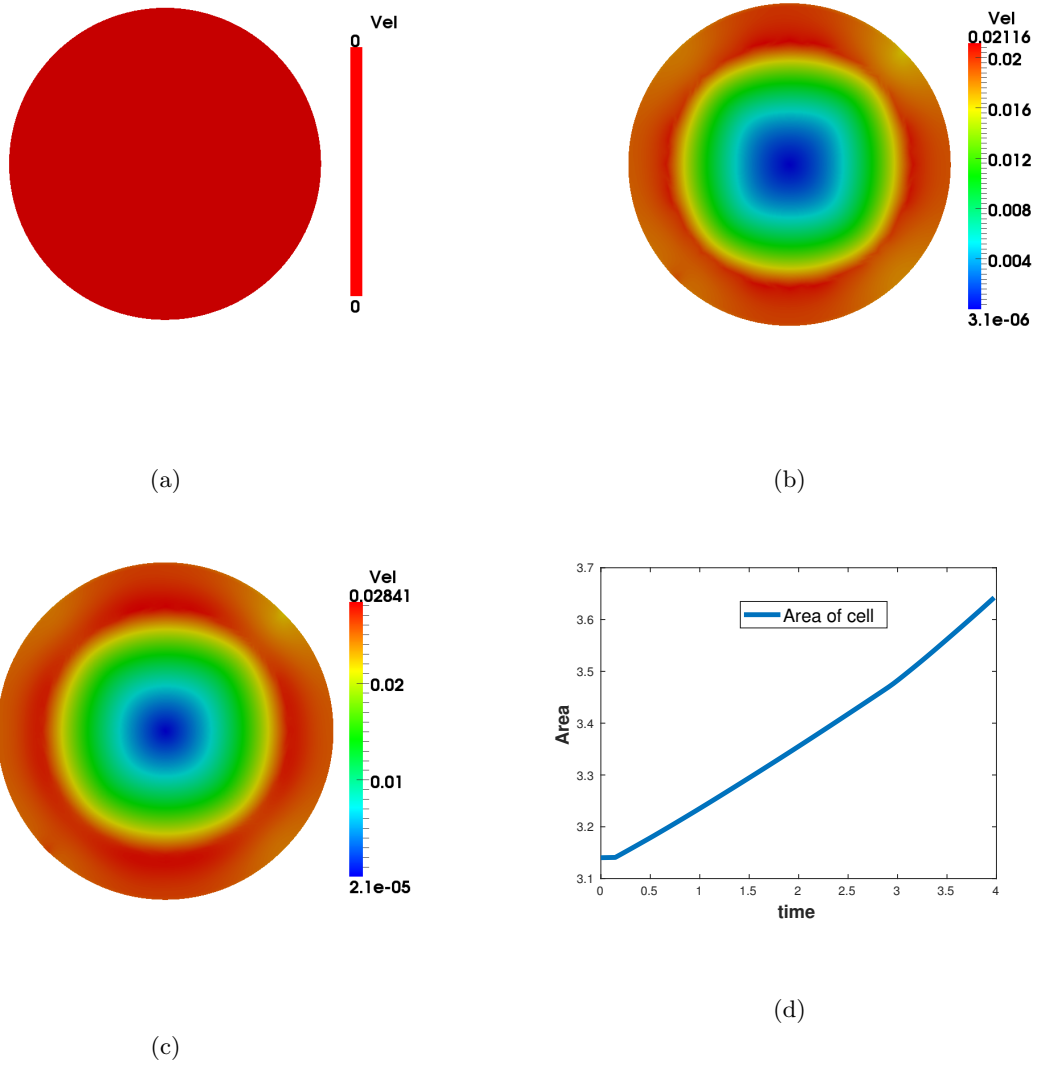


Figure 3.19: Graphical display of speed of the cell as the solution of the force balance equation for the cell. (a) Cell at initial stationary state, (b) cell speed at time  $t = 2$  and (c) cell speed at final time  $t = 4$  and (d) area of the evolving cell as a function of time showing increase in area of the cell with time.

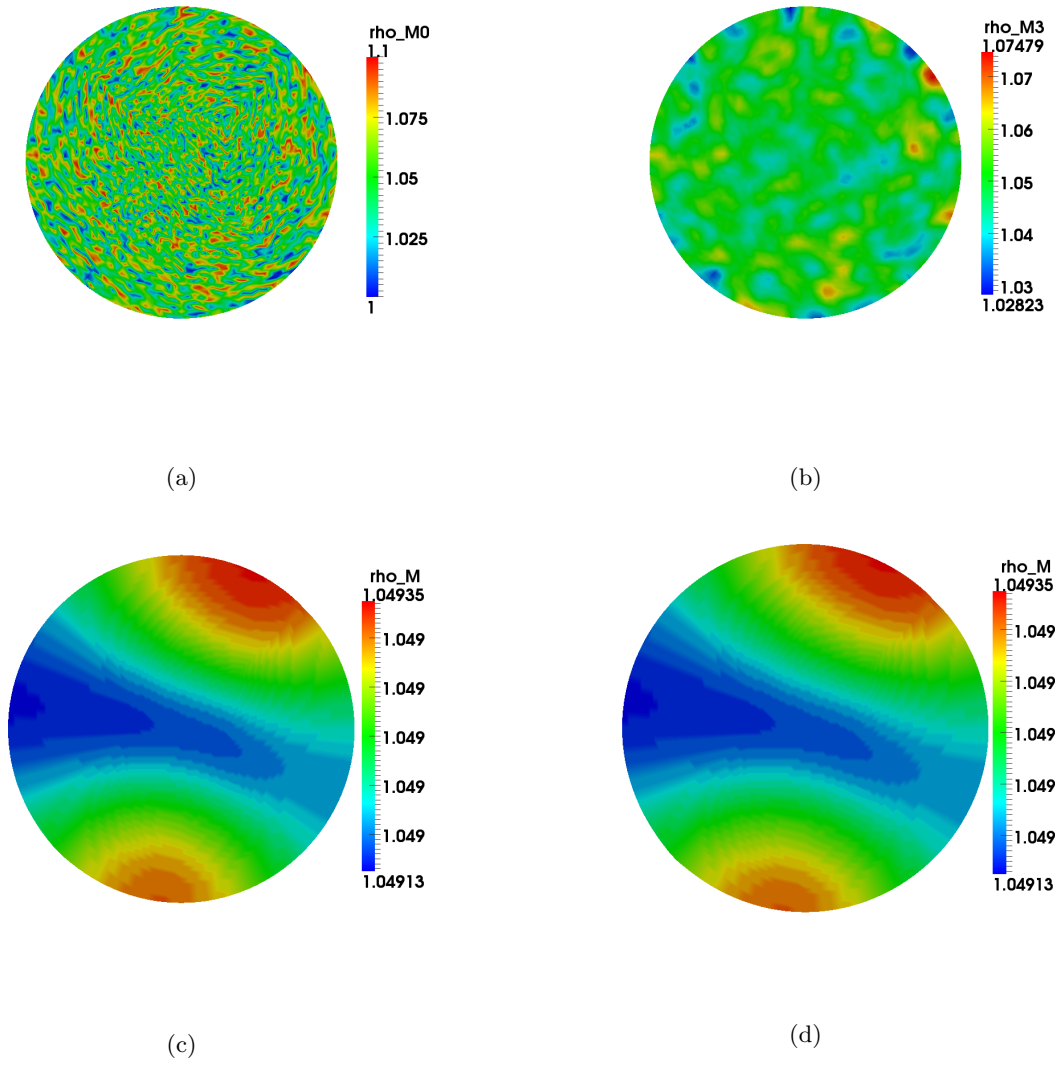


Figure 3.20: Solutions for the  $\rho_m$  variable using  $\tau = 0.001$ . Blue signifies lowest values while red highest values. (a) Initial condition as random perturbation about  $\rho_m = 1$  (b)  $\rho_m$  at time  $t = 0.004$ , (c)  $\rho_m$  at time  $t = 2$  and (d) solution at final time  $t = 4$ .



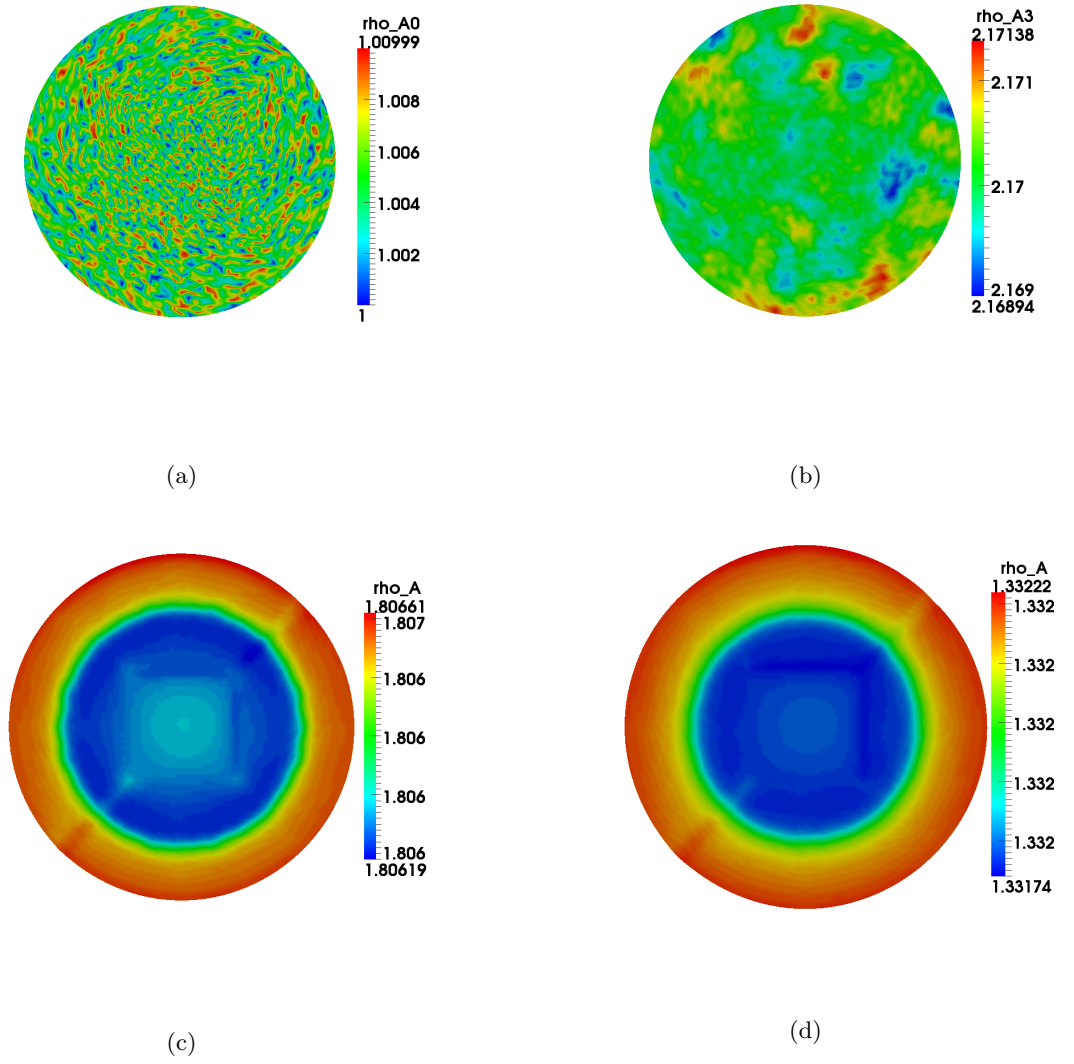


Figure 3.21: Graphical displays of the numerical results of the F-actin concentration  $\rho_a$  with increased total amount of actin  $\rho_a^{tot} = 16$  using a 2-SBDF scheme with  $\tau = 0.001$  and all other parameters held constant. Blue signifies lowest values while red highest values. (a) Initial condition as random perturbation about  $\rho_a = 1.0$  (b)  $\rho_a$  at time  $t = 0.004$ , (c)  $\rho_a$  at time  $t = 2$  and (d) solution at final time  $t = 4$ .

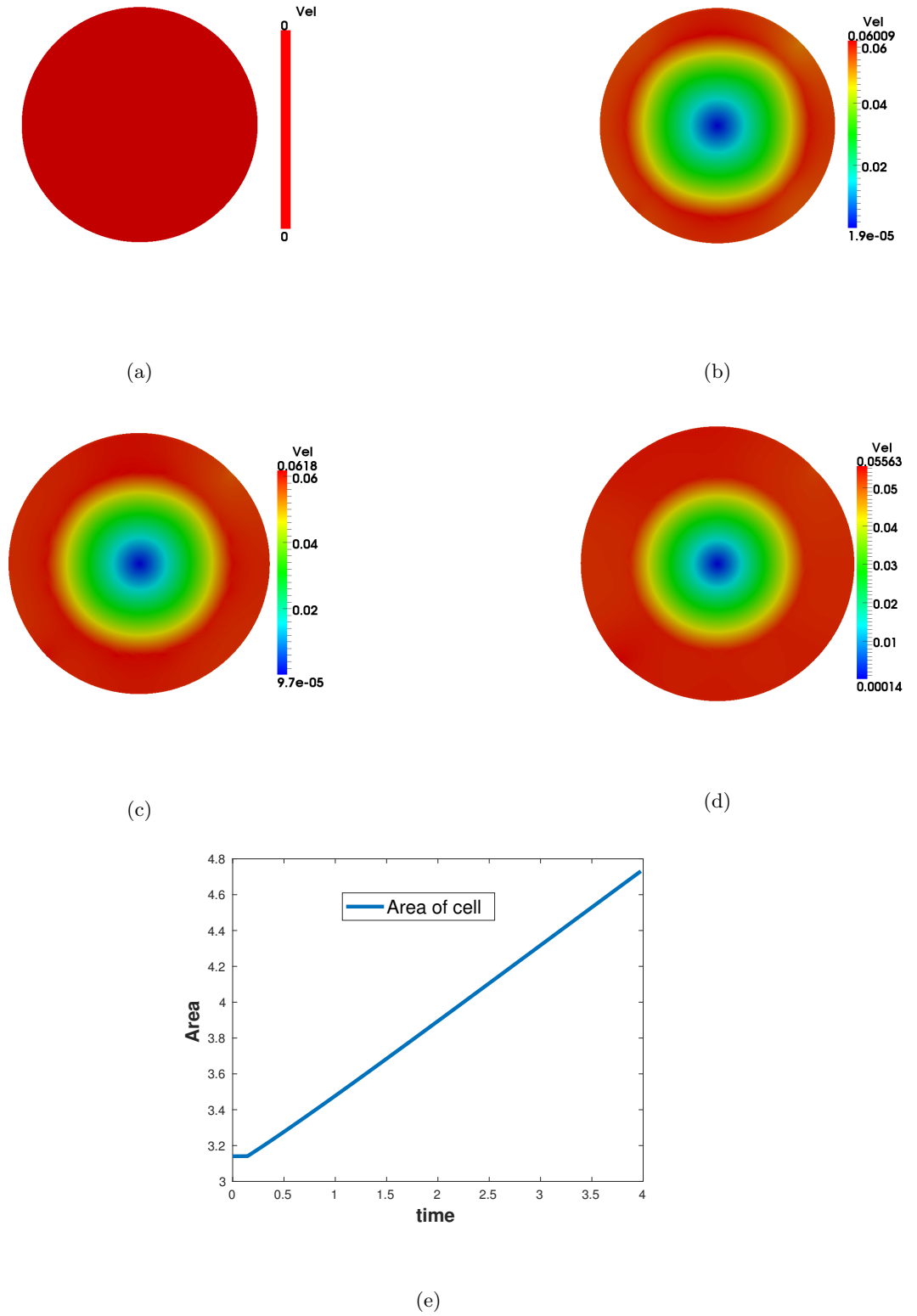


Figure 3.22: Speed of the cell as the solution of the force balance equation for the cell. (a) Cell at initial stationary state (b) cell speed at time  $t = 0.004$ , (c) cell speed at time  $t = 2$ , (d) cell speed at final time  $t = 4$  and (e) area of the evolving cell as a function of time showing increase in area of the cell with time with increased total amount of actin  $\rho_a^{tot} = 16$ .

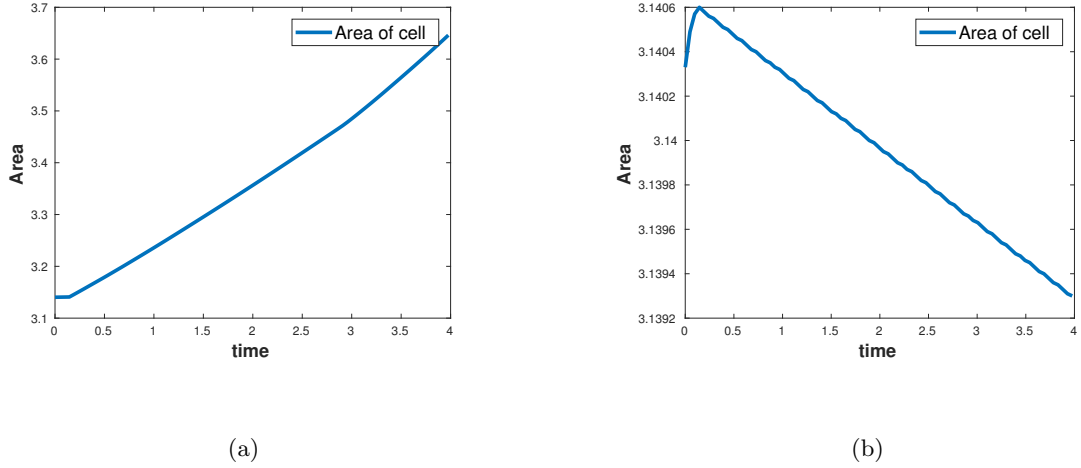


Figure 3.23: Area of the evolving cell as a function of time. (a) area of the cell with contraction coefficient  $\eta_1 = 0.2$  while all other parameters held constant and (b) decreasing area of the cell with a reduced total amount of actin  $\rho_a^{tot} = 5$  while all other parameters held constant.

### Discussion of the numerical results

We have illustrated the numerical solution for the non-dimensionalised viscous model for cell migration which consists of reaction-advection-diffusion equations for F-actin and myosin II and a force balance mechanical equation. The variables for the model are F-actin and myosin II concentrations and flow velocity of the actin-myosin system. We observe that our domain which represents the cell evolved in shape and size with time and the concentrations for myosin II and F-actin reached in-homogeneous steady states. Unlike the biochemical model that we solved earlier, these concentrations do not reach homogeneous steady state. This is because in addition to diffusion, we now have advection of the two quantities following evolution of the cell. We note that the advection term drives cell movement. As F-actin and myosin II assemble inside the cell, they lead to polymerisation and contractile stresses inside the cell. The polymerisation stress causes protrusion forces which extend the cell forward in the region with high F-actin concentration. On the other hand, contractile stress results in cell being pulled inwards isotropically. These two forces play a major role in cell movement. We note that the initial conditions we start with determine greatly the evolution and movement of the cell. Taking a random perturbation about the steady states for F-actin and myosin II resulted in the cell expanding uniformly at the cell periphery while choosing a nonzero F-actin concentration only in one region of the cell resulted in symmetry breaking where the cell determined its front and back and was able to attain a directional movement. Another important parameter in our model is the total amount of actin, i.e, the total F-actin and G-actin concentrations. By varying the parameter for total amount of actin, we arrived at different profiles for the solution. Other parameters that determines the profile of the solution are contraction and polymerisation coefficients. We also plotted graphs which show increase (and decrease) of cell area.

## Summary

In summary, we have solved the non-dimensionalised viscous model (2.3.28) for cell migration. The model comprises of reaction-advection-diffusion equations for actin and myosin II coupled to a force balance equation. We began by deriving the weak formulation of the model followed by space discretisation to arrive at semi-discrete equations. In order to get fully discrete equations, we discretised the time derivative. The fully discrete equations were solved to get solution in terms of concentrations of F-actin and myosin II and speeds of the cell at various points in the cell. Furthermore, we varied some parameters (total amount of actin, contraction coefficient and initial conditions) and obtained different profiles.

In our implementation of the viscous model, we specified a way to update the mesh and we had to keep track of the boundary at every time-step. To avoid this, phase-field methodology can be used. A phase-field function is introduced to distinguish the inside of the cell from the outside. This method is able to find cell shapes without the need of tracking the evolving boundary and has been applied in different modeling aspects [Shao et al. \(2010\)](#). We aim to solve the phase-field model as an alternative model using finite differences in the next section.

## 3.3 Finite differences for the phase-field model

### 3.3.1 Theory of finite difference method

We begin by summarising the theory of finite differences for partial differential equations of the form

$$u_t = \mathcal{L}u, \quad (3.3.1)$$

where  $\mathcal{L}$  is a differential operator in two dimensions which is nonlinear in general containing  $u$  and its spatial derivatives. Finite differences involves subdividing a given domain into sub-domains with grid spacing say  $h_x$  in the  $x$  direction and grid spacing  $h_y$  in the  $y$  direction, approximate the derivatives with finite differences at grid points and seek solutions only at the grid points. We denote any point in the grid by  $\mathbf{x}_{i,j} = (x_i, y_j)$  where  $i, j$  denote indices in the  $x$  and  $y$  directions respectively while solution at any grid point is denoted by  $u_{i,j} = u(x_i, y_j)$ . Below, we summarise the finite differences formulae in two dimensions following [Morton and Mayers \(1998\)](#); [Jain \(1983\)](#) that will be used throughout this section.

#### Forward difference

The forward difference for the gradient of  $u$  is given by

$$\left( \frac{\partial u}{\partial x} \right)_{i,j} = \frac{u_{i+1,j} - u_{i,j}}{h_x} + \mathcal{O}(h_x), \quad (3.3.2)$$

$$\left( \frac{\partial u}{\partial y} \right)_{i,j} = \frac{u_{i,j+1} - u_{i,j}}{h_y} + \mathcal{O}(h_y). \quad (3.3.3)$$

### Backward difference

The backward difference for the gradient of  $u$  is given by

$$\left(\frac{\partial u}{\partial x}\right)_{i,j} = \frac{u_{i,j} - u_{i-1,j}}{h_x} + \mathcal{O}(h_x), \quad (3.3.4)$$

$$\left(\frac{\partial u}{\partial y}\right)_{i,j} = \frac{u_{i,j} - u_{i,j-1}}{h_y} + \mathcal{O}(h_y). \quad (3.3.5)$$

Therefore, both the forward and backward differences are first order methods in space.

### Central difference operator

The second order central difference for the gradient of  $u$  is written as

$$\left(\frac{\partial u}{\partial x}\right)_{i,j} = \frac{u_{i+1,j} - u_{i-1,j}}{2h_x} + \mathcal{O}(h_x^2), \quad (3.3.6)$$

$$\left(\frac{\partial u}{\partial y}\right)_{i,j} = \frac{u_{i,j+1} - u_{i,j-1}}{2h_y} + \mathcal{O}(h_y^2). \quad (3.3.7)$$

while the second order central difference for the laplacian of  $u$  is given by

$$(\Delta u)_{i,j} = \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h_x^2} + \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{h_y^2} + \mathcal{O}(h_x^2, h_y^2). \quad (3.3.8)$$

Next, we consider the time discretisation of the PDE (3.3.1). We subdivide the time interval  $(0, T]$  into  $N$  equal subintervals of size  $\tau = \frac{T}{N}$ . There are two main approaches to time-stepping, namely explicit and implicit methods. The simplest example of an explicit method is the forward Euler method which we define as follows

### Forward Euler method

$$\frac{\partial u^n}{\partial t} = \frac{u^{n+1} - u^n}{\tau} + \mathcal{O}(\tau), \quad (3.3.9)$$

with  $u$  evaluated at point  $(x_i, y_j)$ . Therefore, the forward Euler scheme for (3.3.1) is written as

$$\frac{u^{n+1} - u^n}{\tau} = \mathcal{L}u^n. \quad (3.3.10)$$

This method is conditionally stable Morton and Mayers (1998) in that it requires very small time steps if the mesh size is moderately small. This condition can be very restrictive especially in high dimensional problems. To avoid such restrictions, implicit methods are a natural choice. Examples of implicit methods are the backward Euler method and the trapezoidal rule.

### Backward Euler method

$$\frac{\partial u^n}{\partial t} = \frac{u^n - u^{n-1}}{\tau} + \mathcal{O}(\tau), \quad (3.3.11)$$

with  $u$  evaluated at point  $(x_i, y_j)$ . Therefore, the backward Euler scheme for (3.3.1) is written as

$$\frac{u^{n+1} - u^n}{\tau} = \mathcal{L}u^{n+1}. \quad (3.3.12)$$

### Trapezoidal rule

The trapezoidal rule takes the form

$$\frac{u^{n+1} - u^n}{\tau} = \frac{1}{2} \mathcal{L}(u^n + u^{n+1}), \quad (3.3.13)$$

which is a second order method in time. We note that both the forward Euler and backward Euler are first order methods in time. Now that we have stated the finite difference formulae, we go ahead and solve the phase-field model for cell migration. We begin by rewriting the model equations.

### The non-dimensionalised phase-field model for cell migration

$$\begin{cases} \frac{\partial \phi}{\partial t} + b \nabla \cdot (\phi \beta) = \gamma \Delta \phi - \lambda G'(\phi) + \gamma c |\nabla \phi|, & \mathbf{x} \in \Omega, \quad t \in (0, T] \\ \frac{\partial(\phi \rho_m)}{\partial t} + b \nabla \cdot (\phi \rho_m \beta) = \nabla \cdot \left( \left( \frac{\phi}{1 + a \rho_a} \right) \nabla \rho_m \right), & \mathbf{x} \in \Omega, \quad t \in (0, T], \\ \frac{\partial(\phi \rho_a)}{\partial t} + b \nabla \cdot (\phi \rho_a \beta) = d \nabla \cdot (\phi \nabla \rho_a) + \phi \left( \left( \frac{k_3 \rho_a^2}{1 + \rho_a^2} + k_4 \right) \rho_a^{cyl} - e \rho_a \right), & \mathbf{x} \in \Omega, \quad t \in (0, T], \\ \nabla \cdot (\sigma_\nu(\mathbf{x}, t) + \sigma_{myo}(\mathbf{x}, t) + \sigma_{poly}(\mathbf{x}, t)) = \mathbf{0}, & \mathbf{x} \in \Omega, \quad t \in (0, T], \\ \phi \rho_a(\mathbf{x}, 0) = \phi \rho_a^0(\mathbf{x}), \quad \phi \rho_m(\mathbf{x}, 0) = \phi \rho_m^0(\mathbf{x}), & \mathbf{x} \in \Omega, \quad t = 0, \end{cases} \quad (3.3.14)$$

where

$$\rho_a^{cyl}(t) = \frac{\rho_a^{tot} - \int_{\Omega} \phi \rho_a d\Omega}{\int_{\Omega} \phi d\Omega},$$

$$G(\phi) = 18\phi^2(1 - \phi)^2,$$

$$c = \nabla \cdot \frac{\nabla \phi}{|\nabla \phi|},$$

$$\sigma_\nu(\mathbf{x}, t) = \phi(\nabla \beta(\mathbf{x}, t) + (\nabla \beta(\mathbf{x}, t))^T), \quad \nu_0 \in \mathbb{R}^+,$$

$$\sigma_{myo}(\mathbf{x}, t) = \eta_3 \phi \rho_m(\mathbf{x}, t) \mathbf{I}, \quad \eta_3 \in \mathbb{R}^+,$$

$$\sigma_{poly}(\mathbf{x}, t) = -\eta_4 \phi |\nabla \phi|^2 \rho_a(\mathbf{x}, t) \mathbf{I}, \quad \eta_4 \in \mathbb{R}^+.$$

$a$	$b$	$d$	$e$	$k_3$	$k_4$	$\rho_a^{tot}$	$\lambda$	$\gamma$	$\eta_3$	$\eta_4$
2	500	0.4	500	500	5	8	10	0.4	$\frac{1}{50}$	$\frac{112}{50000}$

Table 3.5: The nondimensionalised values of the parameters used in the model.

### 3.3.2 Finite differences for the phase-field model for cell migration

We let  $\Omega = [-3, 3] \times [-3, 3]$  be the domain and  $\mathcal{I} = (0, T]$  be the time interval. We subdivide the square domain into  $N_\Omega$  uniform grids of grid size  $h_x = h_y = h = \frac{6}{N_\Omega}$  in each direction and divide the time interval  $\mathcal{I} = (0, T]$  into sub-intervals with fixed time step  $\tau$ . We begin with initial conditions  $\phi^{(0)}$ ,  $\rho_m^{(0)}$ ,  $\rho_a^{(0)}$  and  $\beta^{(0)}$  and denote solutions at time  $t^{n+1}$  by  $\phi^{(n+1)}$ ,  $\rho_m^{(n+1)}$ ,  $\rho_a^{(n+1)}$  and

$\beta^{(n+1)} = (\beta_1^{(n+1)}, \beta_2^{(n+1)})$ . The derivatives will be approximated by finite differences as illustrated above.

### Discretisation of the phase-field equation

We begin by discretisation of the phase-field equation

$$\frac{\partial \phi}{\partial t} + b \nabla \cdot (\phi \beta) = \gamma \Delta \phi - \lambda G'(\phi) + \gamma c |\nabla \phi|.$$

We employ time discretisation of the form

$$\frac{\phi^{n+1} - \phi^n}{\tau} = -b \nabla \cdot (\phi^n \beta^{(n)}) + \gamma \Delta \phi^{n+1} - \lambda G'(\phi^n) + \gamma c^{(n)} |\nabla \phi^n|, \quad (3.3.15)$$

where the Laplacian term is treated implicitly. We discretise the Laplacian, divergence and gradient terms using the central difference formula as follows

$$[\Delta \phi^{(n+1)}]_{ij} = \frac{\phi_{i+1,j}^{(n+1)} - 4\phi_{i,j}^{(n+1)} + \phi_{i-1,j}^{(n+1)} + \phi_{i,j+1}^{(n+1)} + \phi_{i,j-1}^{(n+1)}}{h^2}, \quad (3.3.16)$$

$$[\nabla \cdot (\phi^{(n)} \beta^{(n)})]_{ij} = \frac{\phi_{i+1,j}^{(n)} \beta_{1,i+1,j}^{(n)} - \phi_{i-1,j}^{(n)} \beta_{1,i-1,j}^{(n)}}{2h} + \frac{\phi_{i,j+1}^{(n)} \beta_{2,i,j+1}^{(n)} - \phi_{i,j-1}^{(n)} \beta_{2,i,j-1}^{(n)}}{2h}, \quad (3.3.17)$$

$$[\nabla \phi^{(n)}]_{ij} = \left( \frac{\phi_{i+1,j}^{(n)} - \phi_{i-1,j}^{(n)}}{2h}, \frac{\phi_{i,j+1}^{(n)} - \phi_{i,j-1}^{(n)}}{2h} \right), \quad (3.3.18)$$

and end up with the discrete scheme (3.3.19).

$$\frac{\phi_{i,j}^{n+1} - \phi_{i,j}^n}{\tau} = -b [\nabla \cdot (\phi^n \beta^{(n)})]_{ij} + \gamma [\Delta \phi^{n+1}]_{ij} - \lambda [G'(\phi^n)]_{ij} + \gamma [c^{(n)} |\nabla \phi^n|]_{ij}, \quad (3.3.19)$$

with

$$[G'(\phi^n)]_{ij} = 36\phi_{i,j}^{(n)}(1 - \phi_{i,j}^{(n)})(1 - 2\phi_{i,j}^{(n)}), \quad (3.3.20)$$

for  $i, j = 1, 2, 3, \dots, N_\Omega + 1$ . The curvature term is calculated by

$$c^{(n)} = \nabla \cdot \frac{\nabla \phi^{(n)}}{|\nabla \phi^{(n)}|},$$

when  $|\nabla \phi^{(n)}| > 0.01$ , and set to be zero otherwise. Rearranging (3.3.19) gives

$$\phi_{i,j}^{(n+1)} - \tau \gamma [\Delta \phi^{(n+1)}]_{ij} = \phi_{i,j}^{(n)} - b \tau [\nabla \cdot (\phi^{(n)} \beta^{(n)})]_{ij} + \tau \left( -\lambda [G'(\phi^{(n)})]_{ij} + \gamma [c^{(n)} |\nabla \phi^{(n)}|]_{ij} \right), \quad (3.3.21)$$

for  $i, j = 1, 2, 3, \dots, N_\Omega + 1$ . We use the periodic boundary conditions as follows

$$\begin{cases} \phi_{1,j}^{(n+1)} = \phi_{N_\Omega+1,j}^{(n+1)}, \\ \phi_{i,1}^{(n+1)} = \phi_{i,N_\Omega+1}^{(n+1)}, \\ \phi_{0,j}^{(n+1)} = \phi_{N_\Omega,j}^{(n+1)}, \\ \phi_{i,0}^{(n+1)} = \phi_{i,N_\Omega}^{(n+1)}, \\ \phi_{N_\Omega+2,j}^{(n+1)} = \phi_{2,j}^{(n+1)}, \\ \phi_{i,N_\Omega+2}^{(n+1)} = \phi_{i,2}^{(n+1)}. \end{cases} \quad (3.3.22)$$

Incorporating the periodic boundary conditions and letting  $r = \frac{\tau}{h^2}$  gives the following system of equations

$$\mathbf{A}_1 \boldsymbol{\alpha}^{(n+1)} = \mathbf{b}_1^{(n)}, \quad (3.3.23)$$

where

$$\mathbf{A}_1 = \begin{pmatrix} \mathbf{A} & \mathbf{B} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{B} \\ \mathbf{B} & \mathbf{A} & \mathbf{B} & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{B} & \mathbf{A} & \cdots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & & \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{A} & \mathbf{B} \\ \mathbf{B} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{B} & \mathbf{A} \end{pmatrix}, \quad (3.3.24)$$

with

$$\mathbf{A} = \begin{pmatrix} (1+4r\gamma) & -r\gamma & 0 & \cdots & 0 & -r\gamma \\ -r\gamma & (1+4r\gamma) & -r\gamma & \cdots & 0 & 0 \\ 0 & -r\gamma & (1+4r\gamma) & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & & \\ 0 & 0 & 0 & \cdots & (1+4r\gamma) & -r\gamma \\ -r\gamma & 0 & 0 & \cdots & -r\gamma & (1+4r\gamma) \end{pmatrix}, \quad (3.3.25)$$

and

$$\mathbf{B} = \begin{pmatrix} -r\gamma & 0 & 0 & \cdots & 0 & 0 \\ 0 & -r\gamma & 0 & \cdots & 0 & 0 \\ 0 & 0 & -r\gamma & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & & \\ 0 & 0 & 0 & \cdots & -r\gamma & 0 \\ 0 & 0 & 0 & \cdots & 0 & -r\gamma \end{pmatrix}. \quad (3.3.26)$$

The vector  $\boldsymbol{\alpha}^{(n+1)} = (\phi_{1,1}^{(n+1)}, \phi_{2,1}^{(n+1)}, \phi_{3,1}^{(n+1)}, \dots, \phi_{N_{\Omega}+1, N_{\Omega}+1}^{(n+1)})^T$  is the solution vector at time  $t^{(n+1)}$  and the right-hand side of (3.3.23) is given by

$$\mathbf{b}_1^{(n)} = \begin{pmatrix} \phi_{1,1}^{(n)} - b\tau[\nabla \cdot (\phi^{(n)} \boldsymbol{\beta}^{(n)})]_{1,1} + \tau(-\lambda[G'(\phi^{(n)})]_{1,1} + \gamma[c^{(n)}|\nabla \phi^{(n)}|]_{1,1}) \\ \phi_{2,1}^{(n)} - b\tau[\nabla \cdot (\phi^{(n)} \boldsymbol{\beta}^{(n)})]_{2,1} + \tau(-\lambda[G'(\phi^{(n)})]_{2,1} + \gamma[c^{(n)}|\nabla \phi^{(n)}|]_{2,1}) \\ \vdots \\ \phi_{N_{\Omega}+1, N_{\Omega}+1}^{(n)} - b\tau[\nabla \cdot (\phi^{(n)} \boldsymbol{\beta}^{(n)})]_{N_{\Omega}+1, N_{\Omega}+1} + \tau(-\lambda[G'(\phi^{(n)})]_{N_{\Omega}+1, N_{\Omega}+1} + \gamma[c^{(n)}|\nabla \phi^{(n)}|]_{N_{\Omega}+1, N_{\Omega}+1}) \end{pmatrix}$$

with the finite differences as given above.

### Discretisation of the reaction-diffusion equations for myosin and F-actin

We first note that we can write

$$\begin{aligned} \nabla \cdot (\phi D_m(\rho_a) \nabla \rho_m) &= \nabla(\phi D_m(\rho_a)) \cdot \nabla \rho_m + \phi D_m(\rho_a) \Delta \rho_m, \\ \nabla \cdot (\phi \nabla \rho_a) &= \nabla \phi \cdot \nabla \rho_a + \phi \Delta \rho_a. \end{aligned}$$



We apply a forward Euler method to the reaction-advection-diffusion equations for myosin II and F-actin and have

$$\begin{aligned} \frac{\phi^{(n+1)} \rho_m^{(n+1)} - \phi^{(n)} \rho_m^{(n)}}{\tau} &= \nabla(D_m^{(n)} \phi^{(n)}) \cdot \nabla \rho_m^{(n)} + \phi^{(n)} D_m^{(n)} \Delta \rho_m^{(n)} \\ &\quad - b \nabla \cdot (\phi^{(n)} \rho_m^{(n)} \beta^{(n)}), \end{aligned} \quad (3.3.27)$$

and

$$\begin{aligned} \frac{\phi^{(n+1)} \rho_a^{(n+1)} - \phi^{(n)} \rho_a^{(n)}}{\tau} &= D_a (\nabla \phi^{(n)} \cdot \nabla \rho_a^{(n)} + \phi^{(n)} \Delta \rho_a^{(n)}) \\ &\quad - b \nabla \cdot (\phi^{(n)} \rho_a^{(n)} \beta^{(n)}) + \phi^{(n)} f(\rho_a^{(n)}). \end{aligned} \quad (3.3.28)$$

Next, we use central differences to approximate the spatial derivatives as follows

$$[\Delta \rho_a^{(n)}]_{ij} = \frac{\rho_{a,i+1,j}^{(n)} - 4\rho_{a,i,j}^{(n)} + \rho_{a,i-1,j}^{(n)} + \rho_{a,i,j+1}^{(n)} + \rho_{a,i,j-1}^{(n)}}{h^2}, \quad (3.3.29)$$

$$[\Delta \rho_m^{(n)}]_{ij} = \frac{\rho_{m,i+1,j}^{(n)} - 4\rho_{m,i,j}^{(n)} + \rho_{m,i-1,j}^{(n)} + \rho_{m,i,j+1}^{(n)} + \rho_{m,i,j-1}^{(n)}}{h^2}, \quad (3.3.30)$$

$$\begin{aligned} [\nabla \cdot (\phi^{(n)} \rho_a^{(n)} \beta^{(n)})]_{ij} &= \frac{\phi_{i+1,j}^{(n)} \rho_{a,i+1,j}^{(n)} \beta_{1,i+1,j}^{(n)} - \phi_{i-1,j}^{(n)} \rho_{a,i-1,j}^{(n)} \beta_{1,i-1,j}^{(n)}}{2h} \\ &\quad + \frac{\phi_{i,j+1}^{(n)} \rho_{a,i,j+1}^{(n)} \beta_{2,i,j+1}^{(n)} - \phi_{i,j-1}^{(n)} \rho_{a,i,j-1}^{(n)} \beta_{2,i,j-1}^{(n)}}{2h}, \end{aligned} \quad (3.3.31)$$

$$\begin{aligned} [\nabla \cdot (\phi^{(n)} \rho_m^{(n)} \beta^{(n)})]_{ij} &= \frac{\phi_{i+1,j}^{(n)} \rho_{m,i+1,j}^{(n)} \beta_{1,i+1,j}^{(n)} - \phi_{i-1,j}^{(n)} \rho_{m,i-1,j}^{(n)} \beta_{1,i-1,j}^{(n)}}{2h} \\ &\quad + \frac{\phi_{i,j+1}^{(n)} \rho_{m,i,j+1}^{(n)} \beta_{2,i,j+1}^{(n)} - \phi_{i,j-1}^{(n)} \rho_{m,i,j-1}^{(n)} \beta_{2,i,j-1}^{(n)}}{2h}, \end{aligned} \quad (3.3.32)$$

$$[\nabla D_{m,i,j}^{(n)} \phi^{(n)}]_{ij} = \left( \frac{D_{m,i+1,j}^{(n)} \phi_{i+1,j}^{(n)} - D_{m,i-1,j}^{(n)} \phi_{i-1,j}^{(n)}}{2h}, \frac{D_{m,i,j+1}^{(n)} \phi_{i,j+1}^{(n)} - D_{m,i,j-1}^{(n)} \phi_{i,j-1}^{(n)}}{2h} \right). \quad (3.3.33)$$

This gives the following schemes

$$\begin{aligned} \phi_{i,j}^{(n+1)} \rho_{m,i,j}^{(n+1)} &= \tau ([\nabla(D_m^{(n)} \phi^{(n)}) \cdot \nabla \rho_m^{(n)}]_{ij} + \phi_{i,j}^{(n)} D_{m,i,j}^{(n)} [\Delta \rho_m^{(n)}]_{ij}) \\ &\quad + \phi_{i,j}^{(n)} \rho_{m,i,j}^{(n)} - b \tau [\nabla \cdot (\phi^{(n)} \rho_m^{(n)} \beta^{(n)})]_{ij}, \end{aligned} \quad (3.3.34)$$

and

$$\begin{aligned} \phi_{i,j}^{(n+1)} \rho_{a,i,j}^{(n+1)} &= D_a \tau ([\nabla \phi^{(n)} \cdot \nabla \rho_a^{(n)}]_{ij} + \phi_{i,j}^{(n)} [\Delta \rho_a^{(n)}]_{ij}) \\ &\quad + \phi_{i,j}^{(n)} \rho_{a,i,j}^{(n)} - b \tau [\nabla \cdot (\phi^{(n)} \rho_a^{(n)} \beta^{(n)})]_{ij} + \tau \phi_{i,j}^{(n)} f(\rho_{a,i,j}^{(n)}), \end{aligned} \quad (3.3.35)$$

where

$$D_{m,i,j}^{(n)} = \frac{1}{1 + a \rho_{a,i,j}^{(n)}} \quad \text{and} \quad f(\rho_{a,i,j}^{(n)}) = \left( \frac{k_3 (\rho_{a,i,j}^{(n)})^2}{1 + (\rho_{a,i,j}^{(n)})^2} + k_4 \right) \rho_a^{n,cyt} - e \rho_{a,i,j}^{(n)}. \quad (3.3.36)$$

We use periodic boundary conditions of the form

$$\begin{cases} \rho_{a,1,j}^{(n)} = \rho_{a,N_\Omega+1,j}^{(n)}, \\ \rho_{a,i,1}^{(n)} = \rho_{a,i,N_\Omega+1}^{(n)}, \\ \rho_{a,0,j}^{(n)} = \rho_{a,N_\Omega,j}^{(n)}, \\ \rho_{a,i,0}^{(n)} = \rho_{a,i,N_\Omega}^{(n)}, \\ \rho_{a,N_\Omega+2,j}^{(n)} = \rho_{a,2,j}^{(n)}, \\ \rho_{a,i,N_\Omega+2}^{(n)} = \rho_{a,i,2}^{(n)}, \end{cases} \quad (3.3.37)$$

$$\begin{cases} \rho_{m,1,j}^{(n)} = \rho_{m,N_\Omega+1,j}^{(n)}, \\ \rho_{m,i,1}^{(n)} = \rho_{m,i,N_\Omega+1}^{(n)}, \\ \rho_{m,0,j}^{(n)} = \rho_{m,N_\Omega,j}^{(n)}, \\ \rho_{m,i,0}^{(n)} = \rho_{m,i,N_\Omega}^{(n)}, \\ \rho_{m,N_\Omega+2,j}^{(n)} = \rho_{m,2,j}^{(n)}, \\ \rho_{m,i,N_\Omega+2}^{(n)} = \rho_{m,i,2}^{(n)}. \end{cases} \quad (3.3.38)$$

Equations (3.3.34) and (3.3.35) respectively give rise to the following systems of equations

$$\mathbf{A}_2 \boldsymbol{\omega}^{(n+1)} = \mathbf{b}_2^{(n)} \quad \text{and} \quad \mathbf{A}_3 \boldsymbol{\rho}^{(n+1)} = \mathbf{b}_3^{(n)}, \quad (3.3.39)$$

where  $\mathbf{A}_2 = \mathbf{A}_3$  are given by

$$\mathbf{A}_2 = \mathbf{A}_3 = \begin{pmatrix} \phi_{11}^{(n+1)} & 0 & 0 & \cdots & 0 & 0 \\ 0 & \phi_{21}^{(n+1)} & 0 & \cdots & 0 & 0 \\ 0 & 0 & \phi_{31}^{(n+1)} & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & & \\ 0 & 0 & 0 & \cdots & \phi_{N_\Omega-1,N_\Omega}^{(n+1)} & 0 \\ 0 & 0 & 0 & \cdots & 0 & \phi_{N_\Omega,N_\Omega}^{(n+1)} \end{pmatrix}. \quad (3.3.40)$$

The vectors  $\boldsymbol{\rho}^{(n+1)}$  and  $\boldsymbol{\omega}^{(n+1)}$  are the solution vectors and  $\mathbf{b}_2^{(n)}$  and  $\mathbf{b}_3^{(n)}$  are the right hand sides of (3.3.34) and (3.3.35) respectively with the periodic boundary conditions (3.3.37) and (3.3.38). On the right hand side of (3.3.34) and (3.3.35), we note that we have used a Laplacian term of the form

$$\begin{pmatrix} \mathbf{L} & \mathbf{P} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{P} \\ \mathbf{P} & \mathbf{L} & \mathbf{P} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{P} & \mathbf{L} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & & & \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{P} & \mathbf{L} & \mathbf{P} \\ \mathbf{P} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{P} & \mathbf{L} \end{pmatrix}, \quad (3.3.41)$$

where

$$\mathbf{L} = \begin{pmatrix} -4r\gamma & r\gamma & 0 & \cdots & 0 & 0 & r\gamma \\ r\gamma & -4r\gamma & r\gamma & \cdots & 0 & 0 & 0 \\ 0 & r\gamma & -4r\gamma & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & & & \\ 0 & 0 & 0 & \cdots & r\gamma & -4r\gamma & r\gamma \\ r\gamma & 0 & 0 & \cdots & 0 & r\gamma & -4r\gamma \end{pmatrix}, \quad (3.3.42)$$

and

$$\mathbf{P} = \begin{pmatrix} r\gamma & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & r\gamma & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & r\gamma & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & & & \\ 0 & 0 & 0 & \cdots & 0 & r\gamma & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & r\gamma \end{pmatrix}. \quad (3.3.43)$$

### Discretisation of the actin flow equation

To solve the actin flow equation

$$\nabla \cdot (\boldsymbol{\sigma}_\nu(\mathbf{x}, t) + \boldsymbol{\sigma}_{myo}(\mathbf{x}, t) + \boldsymbol{\sigma}_{poly}(\mathbf{x}, t)) = \mathbf{0}, \quad (3.3.44)$$

we construct its corresponding parabolic problem as follows

$$\frac{\partial \boldsymbol{\beta}}{\partial t} = \nabla \cdot (\boldsymbol{\sigma}_\nu(\mathbf{x}, t) + \boldsymbol{\sigma}_{myo}(\mathbf{x}, t) + \boldsymbol{\sigma}_{poly}(\mathbf{x}, t)). \quad (3.3.45)$$

If the solution converges to a limit as  $t \rightarrow \infty$ , then this limit will be the solution of the force balance equation (3.3.44). We will therefore solve (3.3.45) for multiple iterations until we reach a steady state. We first expand the terms in the force balance equation (3.3.44) as follows:

$$\left\{ \begin{array}{l} \nabla \cdot (\phi(\nabla \boldsymbol{\beta}(\mathbf{x}, t) + (\nabla \boldsymbol{\beta}(\mathbf{x}, t))^T)) = \begin{pmatrix} 2\phi \frac{\partial^2 \beta_1}{\partial x^2} + \phi \frac{\partial^2 \beta_1}{\partial y^2} + 2\frac{\partial \phi}{\partial x} \frac{\partial \beta_1}{\partial x} + \frac{\partial \phi}{\partial y} \left( \frac{\partial \beta_2}{\partial x} + \frac{\partial \beta_1}{\partial y} \right) + \phi \frac{\partial^2 \beta_2}{\partial y \partial x} \\ \phi \frac{\partial^2 \beta_2}{\partial x^2} + 2\phi \frac{\partial^2 \beta_2}{\partial y^2} + 2\frac{\partial \phi}{\partial y} \frac{\partial \beta_2}{\partial y} + \frac{\partial \phi}{\partial x} \left( \frac{\partial \beta_2}{\partial x} + \frac{\partial \beta_1}{\partial y} \right) + \phi \frac{\partial^2 \beta_1}{\partial x \partial y} \end{pmatrix}, \\ \nabla \cdot (\eta_3 \phi \rho_m(\mathbf{x}, t) \mathbf{I}) = \begin{pmatrix} \eta_3 \frac{\partial}{\partial x} (\phi \rho_m) \\ \eta_3 \frac{\partial}{\partial y} (\phi \rho_m) \end{pmatrix}, \\ \nabla \cdot (-\eta_4 \phi |\nabla \phi|^2 \rho_a(\mathbf{x}, t) \mathbf{I}) = \begin{pmatrix} -\eta_4 \frac{\partial}{\partial x} (\phi \rho_a \delta_\phi) \\ -\eta_4 \frac{\partial}{\partial y} (\phi \rho_a \delta_\phi) \end{pmatrix}, \end{array} \right. \quad (3.3.46)$$

where for convenience we have used  $\delta_\phi$  to denote the interface term  $|\nabla \phi|^2$ . Splitting equation (3.3.45) into  $x$  and  $y$  directions give

$$\begin{aligned} \frac{\partial \beta_1}{\partial t} = & 2\phi \frac{\partial^2 \beta_1}{\partial x^2} + \phi \frac{\partial^2 \beta_1}{\partial y^2} + 2\frac{\partial \phi}{\partial x} \frac{\partial \beta_1}{\partial x} + \frac{\partial \phi}{\partial y} \left( \frac{\partial \beta_2}{\partial x} + \frac{\partial \beta_1}{\partial y} \right) + \phi \frac{\partial^2 \beta_2}{\partial y \partial x} \\ & + \eta_3 \frac{\partial}{\partial x} (\phi \rho_m) - \eta_4 \frac{\partial}{\partial x} (\phi \rho_a \delta_\phi), \end{aligned} \quad (3.3.47)$$

and

$$\begin{aligned} \frac{\partial \beta_2}{\partial t} = & \phi \frac{\partial^2 \beta_2}{\partial x^2} + 2\phi \frac{\partial^2 \beta_2}{\partial y^2} + 2\frac{\partial \phi}{\partial y} \frac{\partial \beta_2}{\partial y} + \frac{\partial \phi}{\partial x} \left( \frac{\partial \beta_2}{\partial x} + \frac{\partial \beta_1}{\partial y} \right) + \phi \frac{\partial^2 \beta_1}{\partial x \partial y} \\ & + \eta_3 \frac{\partial}{\partial y} (\phi \rho_m) - \eta_4 \frac{\partial}{\partial y} (\phi \rho_a \delta_\phi), \end{aligned} \quad (3.3.48)$$

respectively. To arrive at the discrete versions of (3.3.47) and (3.3.48), we apply central differences to approximate the space derivatives and use a time discretisation as shown in (3.3.49) and (3.3.50). This implies that at each time step  $n + 1$ , we solve discrete equations (3.3.49) and (3.3.50) until convergence has been achieved. For these equations, we note that we set  $\beta_{1,i,j}^{(k=0)} = \beta_{1,i,j}^{(n)}$  and  $\beta_{2,i,j}^{(k=0)} = \beta_{2,i,j}^{(n)}$  and solve the equations iteratively until convergence is achieved. The converged solution will therefore be the solution at time  $t^{(n+1)}$ , i.e.  $\beta_{1,i,j}^{(n+1)}$  and  $\beta_{2,i,j}^{(n+1)}$ . We also note that we make use of the already computed solutions  $\phi_{i,j}^{(n+1)}$ ,  $\rho_{a,i,j}^{(n+1)}$  and  $\rho_{m,i,j}^{(n+1)}$ . Here  $k$  represents the  $k^{th}$  iteration while  $n$  represents time. Therefore, at each time  $n + 1$ , we apply a number of iterations until the solution at time  $n + 1$  converges.

$$\begin{aligned} \beta_{1,i,j}^{(k+1)} - 2\tau \phi_{i,j}^{(n+1)} \left[ 2\phi \frac{\partial^2 \beta_1^{(k+1)}}{\partial x^2} + \phi \frac{\partial^2 \beta_1^{(k+1)}}{\partial y^2} \right]_{ij} = & \beta_{1,i,j}^{(k)} + 2\tau \left[ \frac{\partial \phi^{(n+1)}}{\partial x} \frac{\partial \beta_1^{(k)}}{\partial x} \right]_{ij} \\ & + \tau \left[ \frac{\partial \phi^{(n+1)}}{\partial y} \left( \frac{\partial \beta_2^{(k)}}{\partial x} + \frac{\partial \beta_1^{(k)}}{\partial y} \right) \right]_{ij} \\ & + \tau \phi_{i,j}^{(n+1)} \left[ \frac{\partial^2 \beta_2^{(k)}}{\partial y \partial x} \right]_{ij} \\ & + \tau \eta_3 \left[ \frac{\partial}{\partial x} (\phi^{(n+1)} \rho_m^{(n+1)}) \right]_{ij} \\ & - \tau \eta_4 \left[ \frac{\partial}{\partial x} (\phi^{(n+1)} \rho_a^{(n+1)} \delta_\phi^{(n+1)}) \right]_{ij}, \end{aligned} \quad (3.3.49)$$

and

$$\begin{aligned} \beta_{2,i,j}^{(k+1)} - 2\tau \phi_{i,j}^{(n+1)} \left[ \phi \frac{\partial^2 \beta_2^{(k+1)}}{\partial x^2} + 2\phi \frac{\partial^2 \beta_2^{(k+1)}}{\partial y^2} \right]_{ij} = & \beta_{2,i,j}^{(k)} + 2\tau \left[ \frac{\partial \phi^{(n+1)}}{\partial y} \frac{\partial \beta_2^{(k)}}{\partial y} \right]_{ij} \\ & + \tau \left[ \frac{\partial \phi^{(n+1)}}{\partial x} \left( \frac{\partial \beta_2^{(k)}}{\partial x} + \frac{\partial \beta_1^{(k)}}{\partial y} \right) \right]_{ij} \\ & + \tau \phi_{i,j}^{(n+1)} \left[ \frac{\partial^2 \beta_1^{(k)}}{\partial x \partial y} \right]_{ij} \\ & + \tau \eta_3 \left[ \frac{\partial}{\partial y} (\phi^{(n+1)} \rho_m^{(n+1)}) \right]_{ij} \\ & - \tau \eta_4 \left[ \frac{\partial}{\partial y} (\phi^{(n+1)} \rho_a^{(n+1)} \delta_\phi^{(n+1)}) \right]_{ij}, \end{aligned} \quad (3.3.50)$$

where

$$\left[ \frac{\partial^2 \beta_1^{(k+1)}}{\partial x^2} \right]_{ij} = \frac{\beta_{1,i+1,j}^{(k+1)} - 2\beta_{1,i,j}^{(k+1)} + \beta_{1,i-1,j}^{(k+1)}}{h^2}, \quad (3.3.51)$$

$$\left[ \frac{\partial^2 \beta_1^{(k+1)}}{\partial y^2} \right]_{ij} = \frac{\beta_{1,i,j+1}^{(k+1)} - 2\beta_{1,i,j}^{(k+1)} + \beta_{1,i,j-1}^{(k+1)}}{h^2}, \quad (3.3.52)$$

$$\left[ \frac{\partial^2 \beta_2^{(k)}}{\partial y \partial x} \right]_{ij} = \frac{\beta_{2,i+1,j+1}^{(k)} - \beta_{2,i+1,j-1}^{(k)} - \beta_{2,i-1,j+1}^{(k)} + \beta_{2,i-1,j-1}^{(k)}}{4h^2}, \quad (3.3.53)$$

$$\left[ \frac{\partial^2 \beta_1^{(k)}}{\partial x \partial y} \right]_{ij} = \frac{\beta_{1,i+1,j+1}^{(k)} - \beta_{1,i+1,j-1}^{(k)} - \beta_{1,i-1,j+1}^{(k)} + \beta_{1,i-1,j-1}^{(k)}}{4h^2}, \quad (3.3.54)$$

and similarly to the terms  $\frac{\partial^2 \beta_2}{\partial x^2}$  and  $\frac{\partial^2 \beta_2}{\partial y^2}$ . All the other derivatives can be obtained as before. We use periodic boundary conditions of the form

$$\left\{ \begin{array}{l} \beta_{1,1,j}^{(k+1)} = \beta_{1,N_\Omega+1,j}^{(k+1)}, \\ \beta_{1,i,1}^{(k+1)} = \beta_{1,i,N_\Omega+1}^{(k+1)}, \\ \beta_{1,0,j}^{(k+1)} = \beta_{1,N_\Omega,j}^{(k+1)}, \\ \beta_{1,i,0}^{(k+1)} = \beta_{1,i,N_\Omega}^{(k+1)}, \\ \beta_{1,N_\Omega+2,j}^{(k+1)} = \beta_{1,2,j}^{(k+1)}, \\ \beta_{1,i,N_\Omega+2}^{(1)} = \beta_{1,i,2}^{(k+1)}, \end{array} \right. \quad (3.3.55)$$

$$\left\{ \begin{array}{l} \beta_{2,1,j}^{(k+1)} = \beta_{2,N_\Omega+1,j}^{(k+1)}, \\ \beta_{2,i,1}^{(k+1)} = \beta_{2,i,N_\Omega+1}^{(k+1)}, \\ \beta_{2,0,j}^{(k+1)} = \beta_{2,N_\Omega,j}^{(k+1)}, \\ \beta_{2,i,0}^{(k+1)} = \beta_{2,i,N_\Omega}^{(k+1)}, \\ \beta_{2,N_\Omega+2,j}^{(k+1)} = \beta_{2,2,j}^{(k+1)}, \\ \beta_{2,i,N_\Omega+2}^{(k+1)} = \beta_{2,i,2}^{(k+1)}, \end{array} \right. \quad (3.3.56)$$

and have the following system of equations

$$\mathbf{A}_4 \mathbf{U}^{(n+1)} = \mathbf{b}_4^{(n)} \quad \text{and} \quad \mathbf{A}_5 \mathbf{V}^{(n+1)} = \mathbf{b}_5^{(n)}, \quad (3.3.57)$$

where

$$\mathbf{A}_4 = \begin{pmatrix} \mathbf{C}_1 & \mathbf{D}_1 & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{D}_1 \\ \mathbf{D}_2 & \mathbf{C}_2 & \mathbf{D}_2 & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_3 & \mathbf{C}_3 & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & & & \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{D}_{N_\Omega-1} & \mathbf{C}_{N_\Omega-1} & \mathbf{D}_{N_\Omega-1} \\ \mathbf{D}_{N_\Omega} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{D}_{N_\Omega} & \mathbf{C}_{N_\Omega} \end{pmatrix}, \quad (3.3.58)$$

with

$$\mathbf{C}_j = \begin{pmatrix} 1 + 6r\phi_{1,j}^{(n+1)} & -2r\phi_{1,j}^{(n+1)} & 0 & \cdots & 0 & -2r\phi_{1,j}^{(n+1)} \\ -2r\phi_{2,j}^{(n+1)} & 1 + 6r\phi_{2,j}^{(n+1)} & -2r\phi_{2,j}^{(n+1)} & \cdots & 0 & 0 \\ 0 & -2r\phi_{3,j}^{(n+1)} & 1 + 6r\phi_{3,j}^{(n+1)} & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & & \\ 0 & 0 & 0 & \cdots & 1 + 6r\phi_{N_\Omega-1,j}^{(n+1)} & -2r\phi_{N_\Omega-1,j}^{(n+1)} \\ -2r\phi_{N_\Omega,j}^{(n+1)} & 0 & 0 & \cdots & -2r\phi_{N_\Omega,j}^{(n+1)} & 1 + 6r\phi_{N_\Omega,j}^{(n+1)} \end{pmatrix}$$

and

$$\mathbf{D}_j = \begin{pmatrix} -r\phi_{1,j}^{(n+1)} & 0 & 0 & \cdots & 0 & 0 \\ 0 & -r\phi_{2,j}^{(n+1)} & 0 & \cdots & 0 & 0 \\ 0 & 0 & -r\phi_{3,j}^{(n+1)} & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & & \\ 0 & 0 & 0 & \cdots & -r\phi_{N_\Omega-1,j}^{(n+1)} & 0 \\ 0 & 0 & 0 & \cdots & 0 & -r\phi_{N_\Omega,j}^{(n+1)} \end{pmatrix},$$

while

$$\mathbf{A}_5 = \begin{pmatrix} \mathbf{E}_1 & \mathbf{F}_1 & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{F}_1 \\ \mathbf{F}_2 & \mathbf{E}_2 & \mathbf{F}_2 & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_3 & \mathbf{E}_3 & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & & & \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{F}_{N_\Omega-1} & \mathbf{E}_{N_\Omega-1} & \mathbf{F}_{N_\Omega-1} \\ \mathbf{F}_{N_\Omega} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{F}_{N_\Omega} & \mathbf{E}_{N_\Omega} \end{pmatrix}, \quad (3.3.59)$$

with

$$\mathbf{E}_j = \begin{pmatrix} 1 + 6r\phi_{1,j}^{(n+1)} & -r\phi_{1,j}^{(n+1)} & 0 & \cdots & 0 & -r\phi_{1,j}^{(n+1)} \\ -r\phi_{2,j}^{(n+1)} & 1 + 6r\phi_{2,j}^{(n+1)} & -r\phi_{2,j}^{(n+1)} & \cdots & 0 & 0 \\ 0 & -r\phi_{3,j}^{(n+1)} & 1 + 6r\phi_{3,j}^{(n+1)} & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & & \\ 0 & 0 & 0 & \cdots & 1 + 6r\phi_{N_\Omega-1,j}^{(n+1)} & -r\phi_{N_\Omega-1,j}^{(n+1)} \\ -r\phi_{N_\Omega,j}^{(n+1)} & 0 & 0 & \cdots & -r\phi_{N_\Omega,j}^{(n+1)} & 1 + 6r\phi_{N_\Omega,j}^{(n+1)} \end{pmatrix}$$

and

$$\mathbf{F}_j = \begin{pmatrix} -2r\phi_{1,j}^{(n+1)} & 0 & 0 & \cdots & 0 & 0 \\ 0 & -2r\phi_{2,j}^{(n+1)} & 0 & \cdots & 0 & 0 \\ 0 & 0 & -2r\phi_{3,j}^{(n+1)} & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & & \\ 0 & 0 & 0 & \cdots & -2r\phi_{N_\Omega-1,j}^{(n+1)} & 0 \\ 0 & 0 & 0 & \cdots & 0 & -2r\phi_{N_\Omega,j}^{(n+1)} \end{pmatrix}.$$

The vectors  $\mathbf{U}^{(n+1)}$  and  $\mathbf{V}^{(n+1)}$  are the solution vectors and  $\mathbf{b}_4^{(n)}$  and  $\mathbf{b}_5^{(n)}$  are the right hand sides of (3.3.49) and (3.3.50). The equations we now seek to solve are

$$\begin{cases} \mathbf{A}_1 \boldsymbol{\alpha}^{(n+1)} = \mathbf{b}_1^{(n)}, \\ \mathbf{A}_2 \boldsymbol{\rho}^{(n+1)} = \mathbf{b}_2^{(n)}, \\ \mathbf{A}_3 \boldsymbol{\omega}^{(n+1)} = \mathbf{b}_3^{(n)}, \\ \mathbf{A}_4 \mathbf{U}^{(n+1)} = \mathbf{b}_4^{(n)}, \\ \mathbf{A}_5 \mathbf{V}^{(n+1)} = \mathbf{b}_5^{(n)}. \end{cases} \quad (3.3.60)$$

The matrices in these systems of equations are large and sparse and can therefore be solved by iterative methods e.g preconditioned conjugate gradient and GMRES methods.

### 3.3.3 Numerical simulation

We will only show preliminary results for the phase-field model. Let  $\boldsymbol{\Omega} = [-3, 3] \times [-3, 3]$  be the domain and the unit disk denote the cell at the initial time. We let  $\phi = 1$  inside the cell and  $\phi = 0$  outside the cell. We also allow  $\phi$  to vary smoothly between  $\phi = 1$  and  $\phi = 0$  across the boundary of the cell. We start with initial conditions as random perturbation about the point  $\phi \rho_m = 1$  for myosin II and consider a non-zero concentration of F-actin only in one half of the cell at initial time. The parameters used are as shown in Table 3.5. The domain is subdivided into 100 equal sub-intervals in each direction. We allowed the simulation to run until  $t = 5$ . All the simulations were carried out in Matlab. Figure 3.24 shows evolution of the cell shape with time.

### Discussion of the numerical result

We have attempted to solve the phase-field model as an alternative method for solving model for cell migration. We note that we have only given preliminary results for the phase-field model, i.e, the results for the cell shape. The disk representing the cell is now embedded into the domain  $\boldsymbol{\Omega} = [-3, 3] \times [-3, 3]$ . Our preliminary result for the phase-field model shows that the cell evolves with time and maintains the value  $\phi = 1$  inside the cell and  $\phi = 0$  outside the cell. The diffuse width of the cell is such that  $\phi$  varies smoothly between  $\phi = 1$  and  $\phi = 0$ .

### 3.3.4 Summary

Here, we considered the phase-field model as an alternative model for cell migration. The main advantage of this method is that it avoids the need of tracking the boundary. The model comprises of reaction-advection-diffusion equations for F-actin and myosin II coupled to a force balance equation. To show evolution of the cell, we use an additional equation for the phase-field. This was done by introducing an auxiliary field  $\phi(\mathbf{x}, t)$  which takes the value  $\phi(\mathbf{x}, t) = 1$  inside the cell and the value  $\phi(\mathbf{x}, t) = 0$  outside the cell and varies smoothly between 0 and 1 in the interface between the two regions, a free energy functional  $F[\phi]$  and a double well potential whose minima describes the value of  $\phi$  in each region. We discretised the space using central differences and

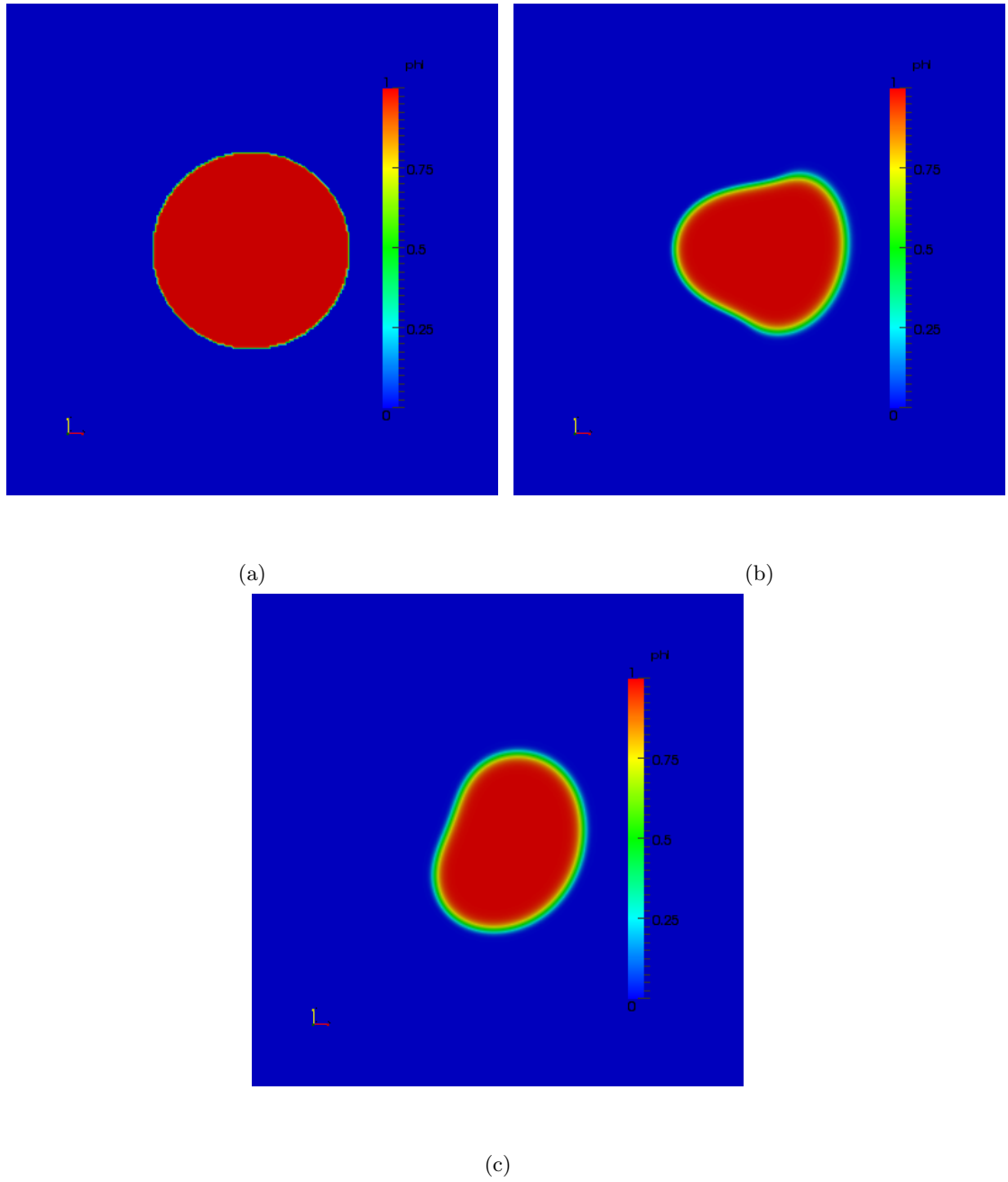


Figure 3.24: Cell shapes at different times showing the cell evolving with time. Red indicates highest values and blue lowest values. (a) cell shape at initial time, (b) cell shape at time  $t = 2$  and (c) cell shape at time  $t = 5$

considered forward and backward Euler methods to discretise the time and ended up with systems of linear equations. The systems of equations were then solved iteratively in matlab to obtain the solution for the phase-field model. We have only displayed preliminary results for this model.

In Chapter 4, we will give a summary, discuss the findings from our work and outline future directions.



## Chapter 4

# Summary and future directions

In this chapter, we summarise and discuss the general results from our work and outline future research.

### 4.1 Summary

Cell migration has become one of the areas of much interest to many researchers. It is a fundamental process in many biological systems, for example wound healing, development of embryos, inflammation, cancer invasion, physiological process among others. In this work, we proposed a model and implemented numerical methods to study cell migration.

We began with a detailed review on the biology of cell migration and outlined various modelling strategies that have been applied to study cell migration. The cell's cytoskeleton is largely responsible for force generation that drive cell migration. It does so through the actin-myosin system. Rapid polymerisation of actin network at the cell periphery leads to the growth of lamellipodia. This leads to expansion of the plasma membrane and thus to the development of a contact area with the substrate. Also, the development of stress fibres and networks that are contractile due to the action of myosin II that tend to slide actin filaments relative to each other leads to contraction of the cell. These active forces from polymerisation of actin and contraction of stress fibres are eventually transmitted to the substrates thereby providing the necessary forces required for cell propulsion. Actin filament and myosin II are therefore responsible for force generations that drive the cell forward. We therefore derived models based on the idea that F-actin and myosin II are the main active stresses in the cell.

We constructed two models for cell migration. The first model is the viscous model which treats the actin-myosin system as a viscous material. This model is composed of reaction-advection-diffusion equations coupled to a force balance mechanical equation. The reaction-advection-diffusion equations for actin and myosin II are derived from conservation laws. The reaction kinetics for actin depends on the concentration of F-actin and actin monomers. We assumed that the total amount of actin is conserved in the entire cell. Actin converts from its active state to

inactive state and vice-versa and is independent of myosin II concentration. Since actin monomers are small molecules and they can diffuse fast, we assumed that they are well mixed inside the cell. We assumed that myosin II only diffuses inside the cell and that the two reaction-advection-diffusion equations are only coupled at the diffusion term for myosin II. For our models, actin and myosin II are the source of active stresses. Therefore at quasi-steady state, all the forces balance off and is equal to zero.

During numerical simulation of the viscous model, we note that we had to specify a way in which the mesh moves and keep track of the boundary at all time steps. To avoid this, one could explore the phase-field framework. We therefore formulated the corresponding phase-field model as an alternative model for cell migration. We introduced a phase-field function  $\phi$  which takes the value  $\phi = 1$  inside the cell and  $\phi = 0$  outside the cell. The phase-field function was allowed to vary smoothly from  $\phi = 1$  to  $\phi = 0$  across the boundary of the cell. We note that this method avoids the need of tracking the boundary every time step. In addition to the reaction-advection-diffusion equations and force balance equation, we now have a phase-field equation which gives the shape of the cell. In both models, we carried out non-dimensionalisation and ended up with dimensionless models. For the viscous model, we proposed to discretise the space using finite elements. The finite element method is well suited for domains that are irregular and evolving. To obtain fully discrete equations, we used the second order semi-implicit backward differentiation formula and solved the model. To begin with, we considered the well studied Schnakenberg model and constructed its numerical solvers. We summarise our observations in the next paragraph.

Actin filaments and myosin II are the main sources of stresses in the cell and are responsible for driving cell migration. We varied the parameter for total actin and observed a linear relationship between the cell expansion and total amount of actin. The more the total amount of actin, the more the expansion. A decrease in the total amount of actin beyond a certain threshold leads to cell shrinking. Myosin II is responsible for cell contraction. By varying the contraction coefficient for myosin II, we observed effect in contraction of the cell. Our results validate the theory that myosin II and actin are the main ingredients that drive cell motility. The initial condition also played a role in the dynamics of cell migration. We considered two sets of initial conditions for the F-actin and myosin II variables. A random perturbation about  $\rho_a = 1$  led to uniform expansion of the cell where the periphery of the cell expanded or contracted uniformly. By considering a non-zero initial concentration of F-actin only in one half of the cell, we observed a directed growth of the cell where the cell expanded in the direction with more concentration of actin and began to migrate in that direction.

The findings and conclusion from our work is therefore as follows: in the absence of advection of actin and myosin II and domain evolution, the biochemical model for F-actin and myosin II will reach steady state. The advection terms drive cell movement. F-actin and myosin II are the main sources of active stresses in the cell and are responsible for driving cell migration. Some of the parameters and variables that are important in the dynamics of cell migration are: the initial conditions for F-actin, the total amount of actin inside the cell, the contraction and polymerisation

coefficients.

For the phase-field model, we proposed to use finite differences to discretise the space. This is because finite differences are simple to work with on simple geometries, which is our case. We note however that on complicated geometries implementation with finite differences is not trivial. We used a forward and backward Euler method to discretise the time derivatives and solved the resulting system of linear equations using iterative methods, for example the preconditioned conjugate gradient and GMRES methods. Our preliminary results for this model was able to give cell shapes at different times. We note that the boundary is now a diffuse layer and the phase-field function distinguishes the inside of the cell from the outside. Our result is able to depict some aspects of cell migration. We note however that some aspects of the results for the phase-field model are missing including the dynamics of F-actin and myosin II and velocity solution.

## 4.2 Future directions

Cell migration is a very complex area and include different modules, for example signalling pathways, actin polymerisation, myosin II contraction, adhesion among others. Our models were able to describe some aspect of cell migration although this is just a few of the many aspects of cell migration. Our model results show consistency with experimental studies but we note that it has several limitations.

- First, the cell is treated as two-dimensional. As much as this is a reasonable approximation for the lamellipodium, it is not sufficient to describe the bulbous cell body. Extension to three dimensions are in principle straightforward, albeit computationally more intensive.
- Secondly, our numerical results indicated expansion of the cell with time. An improvement to the model would be to introduce extra mechanisms for volume conservation which is missing in our models.
- It would be interesting to investigate the effects of different other parameters on cell velocities.
- Furthermore, our reaction-advection-diffusion model can be extended to include more complicated actin-myosin dynamics. For example, it has been suggested that myosin II can disassemble actin filaments at the back of the cell. A modification for the reaction-advection equations would be to introduce a reaction term to the myosin II equation which depends on actin and myosin II and similarly, a reaction term to actin equation that depends on both actin and myosin II.
- Also, it would be interesting to include extra pathways to control the quantity  $\rho_a^{tot}$  so as to enable the cell to polarise.
- It would be interesting to incorporate adhesion mechanism and to investigate the effects of barriers on the cell velocity.

- Finally, an extension of the numerical simulation of the phase-field model in terms of computing the steady state shapes and velocity of the cell and the steady state distributions of F-actin and myosin II can be done in order to compare the two modelling framework.

# Bibliography

Acheson, D. J.

1990. *Elementary fluid dynamics*. Oxford University Press. [15](#), [20](#)

Alberts, B., D. Bray, J. Lewis, M. Raff, K. Roberts, J. D. Watson, and A. Grimstone

1995. Molecular biology of the cell (3rd edn). *Trends in Biochemical Sciences*, 20(5):210–210. [1](#), [2](#)

Alt, W. and M. Dembo

1999. Cytoplasm dynamics and cell motion: two-phase flow models. *Mathematical biosciences*, 156(1-2):207–228. [9](#)

Alt, W. and R. T. Tranquillo

1995. Basic morphogenetic system modeling shape changes of migrating cells: How to explain fluctuating lamellipodial dynamics. *Journal of Biological Systems*, 3(04):905–916. [19](#)

Ananthakrishnan, R. and A. Ehrlicher

2007. The forces behind cell movement. *Int J Biol Sci*, 3(5):303–317. [2](#), [3](#), [4](#), [5](#), [6](#), [7](#), [8](#)

Anderson, K. I. and R. Cross

2000. Contact dynamics during keratocyte motility. *Current biology*, 10(5):253–260. [4](#)

Bangerth, W., R. Hartmann, and G. Kanschat

2007. deal.ii-a general-purpose object-oriented finite element library. *ACM Transactions on Mathematical Software (TOMS)*, 33(4):24. [42](#), [45](#), [54](#), [56](#), [72](#), [75](#)

Barrett, R., M. W. Berry, T. F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. Van der Vorst

1994. *Templates for the solution of linear systems: building blocks for iterative methods*, volume 43. Siam. [45](#), [56](#), [75](#)

Biben, T., K. Kassner, and C. Misbah

2005. Phase-field approach to three-dimensional vesicle dynamics. *Physical Review E*, 72(4):041921. [9](#), [26](#)

Biben, T. and C. Misbah

2003. Tumbling of vesicles under shear flow within an advected-field approach. *Physical Review E*, 67(3):031908. [9](#), [26](#)
- Brebbia, C. A.  
1980. *The boundary element method for engineers*. Pentech press. [10](#), [32](#)
- Buenemann, M., H. Levine, W.-J. Rappel, and L. M. Sander  
2010. The role of cell contraction and adhesion in dictyostelium motility. *Biophysical journal*, 99(1):50–58. [7](#)
- Burridge, K. and E. S. Wittchen  
2013. The tension mounts: stress fibers as force-generating mechanotransducers. *The Journal of cell biology*, 200(1):9–19. [5](#)
- Camley, B. A., Y. Zhao, B. Li, H. Levine, and W.-J. Rappel  
2013. Periodic migration in a physical model of cells on micropatterns. *Physical review letters*, 111(15):158102. [18](#), [19](#), [22](#), [32](#)
- Cavalcanti-Adam, E. A., T. Volberg, A. Micoulet, H. Kessler, B. Geiger, and J. P. Spatz  
2007. Cell spreading and focal adhesion dynamics are regulated by spacing of integrin ligands. *Biophysical journal*, 92(8):2964–2974. [5](#)
- Chung, A. H. W.  
2016. *Novel mathematical and computational approaches for modelling biological systems*. PhD thesis, University of Sussex. [36](#), [45](#)
- Crampin, E. J., E. A. Gaffney, and P. K. Maini  
1999. Reaction and diffusion on growing domains: scenarios for robust pattern formation. *Bulletin of mathematical biology*, 61(6):1093–1120. [32](#)
- Cuvelier, D., M. Théry, Y.-S. Chu, S. Dufour, J.-P. Thiéry, M. Bornens, P. Nassoy, and L. Mahadevan  
2007. The universal dynamics of cell spreading. *Current biology*, 17(8):694–699. [5](#)
- Deckelnick, K., G. Dziuk, and C. M. Elliott  
2005. Computation of geometric partial differential equations and mean curvature flow. *Acta numerica*, 14:139–232. [25](#)
- Du, Q., C. Liu, R. Ryham, and X. Wang  
2005. A phase field formulation of the willmore problem. *Nonlinearity*, 18(3):1249. [25](#), [26](#)
- Dziuk, G. and C. M. Elliott  
2007. Finite elements on evolving surfaces. *IMA journal of numerical analysis*, 27(2):262–292. [66](#)
- Dziuk, G. and C. M. Elliott  
2013. Finite element methods for surface pdes. *Acta Numerica*, 22:289–396. [25](#)

Edelstein-Keshet, L.

1988. *Mathematical models in biology*, volume 46. Siam. [16](#), [38](#)

Fidler, I. J.

2003. The pathogenesis of cancer metastasis: the 'seed and soil' hypothesis revisited. *Nature Reviews Cancer*, 3(6):453. [8](#)

Fournier, M. F., R. Sauser, D. Ambrosi, J.-J. Meister, and A. B. Verkhovsky

2010. Force transmission in migrating cells. *The Journal of cell biology*, 188(2):287–297. [6](#)

Freund, R. W., G. H. Golub, and N. M. Nachtigal

1992. Iterative solution of linear systems. *Acta numerica*, 1:57–100. [56](#), [75](#)

Gelfand, I. M., R. A. Silverman, et al.

2000. *Calculus of variations*. Courier Corporation. [26](#)

George, U. Z.

2012. *A numerical approach to studying cell dynamics*. PhD thesis. [9](#), [20](#), [45](#)

Gilbarg, D. and N. S. Trudinger

2015. *Elliptic partial differential equations of second order*. springer. [14](#), [15](#)

Gracheva, M. E. and H. G. Othmer

2004. A continuum model of motility in ameboid cells. *Bulletin of mathematical biology*, 66(1):167–193. [9](#)

Gupta, V.

2012. Rupture of multiple receptor-ligand bonds: Bimodal distribution of bond rupture force. *EUROPEAN PHYSICAL JOURNAL E*, 35(9). [6](#)

Hall, W. S.

1994. Boundary element method. In *The Boundary Element Method*, Pp. 61–83. Springer. [32](#)

Houston, P., C. Schwab, and E. Süli

2002. Discontinuous hp-finite element methods for advection-diffusion-reaction problems. *SIAM Journal on Numerical Analysis*, 39(6):2133–2163. [10](#)

Jain, M. K.

1983. Numerical solution of differential equations. *JOHN WILEY & SONS, INC., 605 THIRD AVE., NEW YORK, NY 10158, USA, 1983, 510*. [88](#)

Karma, A., D. A. Kessler, and H. Levine

2001. Phase-field model of mode iii dynamic fracture. *Physical Review Letters*, 87(4):045501. [25](#)

Kockelkoren, J., H. Levine, and W.-J. Rappel

2003. Computational approach for modeling intra-and extracellular dynamics. *Physical Review E*, 68(3):037702. [25](#)

Larsson, S. and V. Thomée

2003. Finite difference methods for hyperbolic equations. *Partial Differential Equations with Numerical Methods*, Pp. 185–199. [15](#)

Lauffenburger, D. A. and A. F. Horwitz

1996. Cell migration: a physically integrated molecular process. *Cell*, 84(3):359–369. [8](#)

Lewis, M. and J. Murray

1991. Analysis of stable two-dimensional patterns in contractile cytogel. *Journal of Nonlinear Science*, 1(3):289–311. [19](#), [20](#)

Li, X., J. Lowengrub, A. Rätz, and A. Voigt

2009. Solving pdes in complex geometries: a diffuse domain approach. *Communications in mathematical sciences*, 7(1):81. [25](#)

Liu, P., Y. Zhang, Q. Cheng, and C. Lu

2007. Simulations of the spreading of a vesicle on a substrate surface mediated by receptor–ligand binding. *Journal of the Mechanics and Physics of Solids*, 55(6):1166–1181. [6](#), [9](#)

Lowengrub, J. S., A. Rätz, and A. Voigt

2009. Phase-field modeling of the dynamics of multicomponent vesicles: Spinodal decomposition, coarsening, budding, and fission. *Physical Review E*, 79(3):031926. [9](#)

Madzvamuse, A.

2000. *A numerical approach to the study of spatial pattern formation*. PhD thesis, University of Oxford. [15](#), [16](#), [36](#), [38](#), [45](#), [49](#), [56](#), [75](#)

Madzvamuse, A.

2006. Time-stepping schemes for moving grid finite elements applied to reaction–diffusion systems on fixed and growing domains. *Journal of computational physics*, 214(1):239–263. [10](#), [11](#), [33](#), [36](#), [41](#), [50](#)

Madzvamuse, A. and A. H. Chung

2014. Fully implicit time-stepping schemes and non-linear solvers for systems of reaction–diffusion equations. *Applied Mathematics and Computation*, 244:361–374. [11](#), [33](#), [36](#)

Madzvamuse, A., P. K. Maini, and A. J. Wathen

2005. A moving grid finite element method for the simulation of pattern generation by turing models on growing domains. *Journal of Scientific Computing*, 24(2):247–262. [10](#), [13](#), [16](#), [32](#), [33](#), [36](#)

Madzvamuse, A., A. J. Wathen, and P. K. Maini

2003. A moving grid finite element method applied to a model biological pattern generator. *Journal of computational physics*, 190(2):478–500. [10](#), [33](#)



- Marée, A. F., A. Jilkin, A. Dawes, V. A. Grieneisen, and L. Edelstein-Keshet  
2006. Polarization and movement of keratocytes: a multiscale modelling approach. *Bulletin of mathematical biology*, 68(5):1169–1211. [9](#)
- Mitchell, A. R. and D. F. Griffiths  
1980. *The finite difference method in partial differential equations*, number BOOK. John Wiley. [10](#), [32](#)
- Mitra, S. K., D. A. Hanson, and D. D. Schlaepfer  
2005. Focal adhesion kinase: in command and control of cell motility. *Nature reviews Molecular cell biology*, 6(1):56–68. [7](#)
- Mogilner, A.  
2009. Mathematics of cell motility: have we got its number? *Journal of mathematical biology*, 58(1-2):105. [8](#)
- Mogilner, A. and L. Edelstein-Keshet  
2002. Regulation of actin dynamics in rapidly moving cells: a quantitative analysis. *Biophysical journal*, 83(3):1237–1258. [7](#), [9](#)
- Mogilner, A. and K. Keren  
2009. The shape of motile cells. *Current Biology*, 19(17):R762–R771. [7](#), [8](#)
- Mogilner, A., E. Marland, and D. Bottino  
2001. A minimal model of locomotion applied to the steady gliding movement of fish keratocyte cells. In *Mathematical Models for Biological Pattern Formation*, Pp. 269–293. Springer. [9](#)
- Mogilner, A. and G. Oster  
1996. Cell motility driven by actin polymerization. *Biophysical journal*, 71(6):3030–3045. [8](#)
- Mogilner, A. and G. Oster  
2003. Force generation by actin polymerization ii: the elastic ratchet and tethered filaments. *Biophysical journal*, 84(3):1591–1605. [8](#), [9](#)
- Mori, Y., A. Jilkin, and L. Edelstein-Keshet  
2008. Wave-pinning and cell polarity from a bistable reaction-diffusion system. *Biophysical journal*, 94(9):3684–3697. [18](#)
- Morton, K. and D. Mayers  
1998. Numerical solution of partial differential equations. *Journal of Fluid Mechanics*, 363:349–349. [10](#), [12](#), [32](#), [35](#), [88](#), [89](#)
- Murray, J. D.  
2001. *Mathematical biology. II Spatial models and biomedical applications {Interdisciplinary Applied Mathematics V. 18}*. Springer-Verlag New York Incorporated New York. [16](#), [19](#), [20](#), [37](#), [38](#)

Murray, J. D.

2002. Mathematical biology i: an introduction. [15](#), [16](#), [36](#), [37](#), [38](#), [45](#), [49](#)

Pellegrin, S. and H. Mellor

2007. Actin stress fibres. *Journal of cell science*, 120(20):3491–3499. [4](#), [5](#)

Peskin, C. S., G. M. Odell, and G. F. Oster

1993. Cellular motions and thermal fluctuations: the brownian ratchet. *Biophysical journal*, 65(1):316–324. [8](#)

Pollard, T. D.

2007. Regulation of actin filament assembly by arp2/3 complex and formins. *Annu. Rev. Biophys. Biomol. Struct.*, 36:451–477. [3](#), [4](#), [16](#), [18](#)

Pollard, T. D., L. Blanchoin, and R. D. Mullins

2000. Molecular mechanisms controlling actin filament dynamics in nonmuscle cells. *Annual review of biophysics and biomolecular structure*, 29(1):545–576. [4](#)

Pollard, T. D. and G. G. Borisy

2003. Cellular motility driven by assembly and disassembly of actin filaments. *Cell*, 112(4):453–465. [viii](#), [4](#), [5](#), [8](#), [21](#)

Pullarkat, P. A., P. A. Fernández, and A. Ott

2007. Rheological properties of the eukaryotic cell cytoskeleton. *Physics Reports*, 449(1-3):29–53. [viii](#), [2](#), [3](#), [4](#), [7](#), [8](#), [18](#), [19](#)

Qian, J., J. Wang, Y. Lin, and H. Gao

2009. Lifetime and strength of periodic bond clusters between elastic media under inclined loading. *Biophysical Journal*, 97(9):2438–2445. [6](#)

Rappel, W.-J.

2001. Filament instability and rotational tissue anisotropy: a numerical study using detailed cardiac models. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 11(1):71–80. [25](#)

Rätz, A., A. Voigt, et al.

2006. Pde’s on surfaces—a diffuse interface approach. *Communications in Mathematical Sciences*, 4(3):575–590. [25](#)

Reddy, J. N.

1993. *An introduction to the finite element method*, volume 2. McGraw-Hill New York. [10](#), [12](#), [14](#), [32](#)

Ridley, A. J.

2011. Life at the leading edge. *Cell*, 145(7):1012–1022. [5](#)

- Ridley, A. J., M. A. Schwartz, K. Burridge, R. A. Firtel, M. H. Ginsberg, G. Borisy, J. T. Parsons, and A. R. Horwitz  
2003. Cell migration: integrating signals from front to back. *Science*, 302(5651):1704–1709. [8](#)
- Rubinstein, B., M. F. Fournier, K. Jacobson, A. B. Verkhovsky, and A. Mogilner  
2009. Actin-myosin viscoelastic flow in the keratocyte lamellipod. *Biophysical Journal*, 97(7):1853–1863. [19](#)
- Rubinstein, B., K. Jacobson, and A. Mogilner  
2005. Multiscale two-dimensional modeling of a motile simple-shaped cell. *Multiscale Modeling & Simulation*, 3(2):413–439. [9](#)
- Ruuth, S. J.  
1993. *Implicit-explicit methods for time-dependent PDEs*. PhD thesis, University of British Columbia. [11](#)
- Ruuth, S. J.  
1995. Implicit-explicit methods for reaction-diffusion problems in pattern formation. *Journal of Mathematical Biology*, 34(2):148–176. [10](#), [11](#), [33](#), [41](#), [50](#)
- Saad, Y.  
2003. *Iterative methods for sparse linear systems*, volume 82. siam. [45](#), [56](#), [75](#)
- Satulovsky, J., R. Lui, and Y.-l. Wang  
2008. Exploring the control circuit of cell migration by mathematical modeling. *Biophysical journal*, 94(9):3671–3683. [9](#)
- Satyanarayana, S. and A. Baumgaertner  
2004. Shape and motility of a model cell: a computational study. *The Journal of chemical physics*, 121(9):4255–4265. [9](#)
- Schaub, S., S. Bohnet, V. M. Laurent, J.-J. Meister, and A. B. Verkhovsky  
2007. Comparative maps of motion and assembly of filamentous actin and myosin ii in migrating cells. *Molecular biology of the cell*, 18(10):3723–3732. [7](#)
- Schnakenberg, J.  
1979. Simple chemical reaction systems with limit cycle behaviour. *Journal of theoretical biology*, 81(3):389–400. [36](#)
- Schwarz, U. S. and S. A. Safran  
2013. Physics of adherent cells. *Reviews of Modern Physics*, 85(3):1327. [1](#), [4](#), [5](#), [6](#), [7](#)
- Shao, D.  
2011. Computational models on cell migration. [9](#), [25](#)

Shao, D., H. Levine, and W.-J. Rappel

2012. Coupling actin flow, adhesion, and morphology in a computational cell motility model. *Proceedings of the National Academy of Sciences*, 109(18):6851–6856. [9](#), [22](#), [32](#)

Shao, D., W.-J. Rappel, and H. Levine

2010. Computational model for cell morphodynamics. *Physical review letters*, 105(10):108104. [9](#), [32](#), [88](#)

Shemesh, T., A. D. Bershadsky, and M. M. Kozlov

2012. Physical model for self-organization of actin cytoskeleton and adhesion complexes at the cell front. *Biophysical journal*, 102(8):1746–1756. [5](#), [6](#)

Stephanou, A., M. Chaplain, and P. Tracqui

2004. A mathematical model for the dynamics of large membrane deformations of isolated fibroblasts. *Bulletin of mathematical biology*, 66(5):1119. [9](#), [19](#), [20](#)

Süli, E.

2007. Finite element methods for partial differential equations. *University of Oxford*. [10](#), [32](#), [34](#)

Teigen, K. E., X. Li, J. Lowengrub, F. Wang, and A. Voigt

2009. A diffuse-interface approach for modeling transport, diffusion and adsorption/desorption of material quantities on a deformable interface. *Communications in mathematical sciences*, 4(7):1009. [25](#)

Turing, A. M.

1952. The chemical basis of morphogenesis. *Phil. Trans. R. Soc. Lond. B*, 237(641):37–72. [10](#), [36](#), [37](#), [49](#)

Vogel, V. and M. P. Sheetz

2009. Cell fate regulation by coupling mechanical cycles to biochemical signaling pathways. *Current opinion in cell biology*, 21(1):38–46. [6](#)

Zhang, J., S. Das, and Q. Du

2009. A phase field model for vesicle–substrate adhesion. *Journal of Computational Physics*, 228(20):7837–7849. [9](#)

Zhu, C., G. Bao, and N. Wang

2000. Cell mechanics: mechanical response, cell adhesion, and molecular deformation. *Annual review of biomedical engineering*, 2(1):189–226. [4](#), [5](#)

Zill, D., W. S. Wright, and M. R. Cullen

2011. *Advanced engineering mathematics*. Jones & Bartlett Learning. [76](#)